

RICE UNIVERSITY

**Pseudospectral Collocation Methods for the Direct
Transcription of Optimal Control Problems**

by

Jesse A. Pietz

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

Master of Arts

APPROVED, THESIS COMMITTEE:

Matthias Heinkenschloss, Chairman
Associate Professor of Computational and
Applied Mathematics

William Symes
Noah Harding Professor of Computational
and Applied Mathematics

Yin Zhang
Professor of Computational and Applied
Mathematics

Nazareth Bedrossian
Principal Member of Technical Staff,
Charles Stark Draper Laboratory, Inc.

Houston, Texas

April, 2003

Report Documentation Page			Form Approved OMB No. 0704-0188	
<p>Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p>				
1. REPORT DATE 00 APR 2003	2. REPORT TYPE N/A	3. DATES COVERED -		
Pseudospectral Collocation Methods for the Direct Transcription of Optimal Control Problems			5a. CONTRACT NUMBER	
			5b. GRANT NUMBER	
			5c. PROGRAM ELEMENT NUMBER	
Rice University, Houston, TX			5d. PROJECT NUMBER	
			5e. TASK NUMBER	
			5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)			8. PERFORMING ORGANIZATION REPORT NUMBER	
Rice University, Houston, TX			10. SPONSOR/MONITOR'S ACRONYM(S)	
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited				
13. SUPPLEMENTARY NOTES				
14. ABSTRACT				
15. SUBJECT TERMS				
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 140
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified		

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

ABSTRACT

Pseudospectral Collocation Methods for the Direct Transcription of Optimal Control Problems

by

Jesse A. Pietz

This thesis is concerned with the study of pseudospectral discretizations of optimal control problems governed by ordinary differential equations and with their application to the solution of the International Space Station (ISS) momentum dumping problem.

Pseudospectral methods are used to transcribe a given optimal control problem into a nonlinear programming problem. Adjoint estimates are presented and analyzed that provide approximations of the original adjoint variables using Lagrange multipliers corresponding to the discretized optimal control problem. These adjoint estimations are derived for a broad class of pseudospectral discretizations and generalize the previously known adjoint estimation procedure for the Legendre pseudospectral discretization. The error between the desired solution to the infinite dimensional optimal control problem and the solution computed using pseudospectral collocation and nonlinear programming is estimated for linear-quadratic optimal control problems. Numerical results are given for both linear-quadratic and nonlinear optimal control problems.

The Legendre pseudospectral method is applied to formulations of the ISS momentum dumping problem. Computed solutions are verified through simulations using adaptive higher order integration of the system dynamics.

Acknowledgments

I would like to give special thanks to my advisors Dr. Matthias Heinkenschloss and Dr. Nazareth Bedrossian. Without their guidance and dedication this thesis would not have been possible.

Thanks go to those at Draper Lab who have provided valuable insight along the way. Chris, Ed, Jimmy, Mike, Nathan, I appreciate all your help.

I would like to express my gratitude to the faculty in the CAAM department at Rice University, especially my committee members, who have all contributed to this work in one way or another. Thanks to the administrative staff, Daria and Ginger, for keeping me on my toes. Thanks also go to the graduate students who have been there with me. Agata, Dwayne , Ed, Eddie, Keith, Kyle, Mark, Mike, Weldon, and a possibly others who I cannot think of right now, thanks for the pleasant memories.

Thanks to the Air Force for making this all possible.

To all of my friends and family, thanks for the support. Most of all, I would like to thank my wife (name omitted due to Air Force policy) for all her love and support.

ACKNOWLEDGMENT

April 21, 2003

This thesis was prepared at The Charles Stark Draper Laboratory, Inc., under NAS9-01069, sponsored by NASA Johnson Space Center, Houston, Texas.

Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

(author's signature)

ASSIGNMENT

Draper Laboratory Report Number T-1444.

In consideration for the research opportunity and permission to prepare my thesis by and at The Charles Stark Draper Laboratory, Inc., I hereby assign my copyright of the thesis to The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts.

(author's signature)

April 21, 2003

Table of Contents

Abstract	iii
Acknowledgments	iv
List of Figures	ix
List of Tables	xiv
1 Introduction	1
2 Optimal Control Problems Governed by Ordinary Differential Equations	5
2.1 Problem Statement and Necessary Optimality Conditions	5
2.2 Bolza and Mayer Forms of the Optimal Control Problem	8
2.3 Prototypical Examples	11
3 Direct Transcription of Optimal Control Problems	14
3.1 Pseudospectral Discretization of the Optimal Control Problem	16
3.2 Optimality Conditions for the Discretized Optimal Control Problem .	21
3.3 Pseudospectral Method Adjoint Estimation Properties	22
3.3.1 Integration by Parts Approach	23
3.3.2 Weighting Matrix Minimization Approach	35
3.4 Discretization Error for the Optimal Control	42
3.5 Numerical Equivalence of Bolza and Mayer Forms	49
3.6 Extension to Multiple Subintervals	55
4 International Space Station Momentum Dumping Problem	60
4.1 Background	60
4.2 Rotational Dynamics	62

4.2.1	International Space Station Assembly Stage 12A	65
4.2.2	Boundary Conditions	67
4.3	ISS Momentum Dumping Problem with Continuous Control	69
4.4	ISS Momentum Dumping Problem with Piecewise Constant Control .	77
4.4.1	Constraint on the Control Magnitude	85
4.5	ISS Momentum Dumping Problem with Control Law	92
5	Conclusions and Future Work	102
	Bibliography	104
A	DIDO: A Tool for Direct and InDirect Optimization	110
B	MATLAB Code for the Solution of the ISS Momentum Dumping Problem Using DIDO	113
B.1	ISS Momentum Dumping Problem with Continuous Control	113
B.2	ISS Momentum Dumping Problem with Piecewise Constant Control .	118
B.3	ISS Momentum Dumping Problem with Control Law	123

List of Figures

3.1 Ratio Between Minimum Singular Value and Maximum Singular Value of the Constraint Jacobian at the Solution for Various N . - Left: Linear-Quadratic OCP in Mayer Form - Right: Orbit Transfer OCP	19
3.2 Lagrange Multipliers $\tilde{\lambda}^N$ and Estimated Adjoints $g\lambda^N$. Top Left: Legendre Pseudospectral Lagrange Multipliers $\tilde{\lambda}^N$. Top Right: Chebyshev Pseudospectral Lagrange Multipliers $\tilde{\lambda}^N$. Mid Left: Legendre Pseudospectral Adjoint Estimates. Mid Right: Chebyshev Pseudospectral Adjoint Estimates. Bottom Middle: Error Between Legendre Adjoint Estimates and Chebyshev Adjoint Estimates	34
3.3 The least squares norms $\ \widetilde{W}D + D^T\widetilde{W} - E\ _F^2$ for \widetilde{w} given by (3.47) for different numbers of collocation points	38
3.4 The least squares norms $\ \widetilde{W}D + D^T\widetilde{W} - E\ _F^2$ for different numbers of collocation points	38
3.5 $\min \widetilde{w}_i$ and $\max \widetilde{w}_i$ for different numbers of collocation points	39
3.6 Element Wise Error $\log_{10} (\widetilde{W}D + D^T\widetilde{W} - E)_{ij} $ for $N = 64$	40
3.7 Error vs. N for Linear-Quadratic Optimal Control Problem in Mayer Form Using Legendre Pseudospectral Collocation- Top Left: ℓ_2 State Error - Top Right: ℓ_2 Control Error - Bottom Left: ℓ_2 Adjoint Divided by Weighting Function Error - Bottom Right: Norm of System Matrix Inverse	47

3.8	Error vs. N for Linear-Quadratic Optimal Control Problem in Mayer Form Using Chebyshev Pseudospectral Collocation- Top Left: ℓ_2 State Error - Top Right: ℓ_2 Control Error - Bottom Left: ℓ_2 Adjoint Divided by Weighting Function Error - Bottom Right: Norm of System Matrix Inverse	48
3.9	ℓ_2 Error between Mayer Form Problem and Bolza Form Problem vs. N for Linear-Quadratic Optimal Control Problem Using Legendre Pseudospectral Collocation- Top Left: State Error - Top Right: Control Error - Bottom Middle: Adjoint Divided by Weighting Function Error	54
4.1	Earth's Inertial Reference Frame, Local Vertical Local Horizontal Reference Frame, International Space Station's Body Reference Frame (ISS Assembly 12A shown)	64
4.2	International Space Station Assembly 12A	66
4.3	Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Continuous Control, Using N=150	71
4.4	Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Continuous Control, Using N=150	72
4.5	Computed Optimal Control, Extended by Zero for $t > 1800$, for the Space Station Momentum Dumping Problem with Continuous Control, Using N=150.	73
4.6	Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Continuous Control, Using N=150	74
4.7	Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Continuous Control, Using N=150	75

4.8	Simulated External Torques for Space Station Momentum Dumping Problem with Continuous Control, Using N=150	76
4.9	Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals	79
4.10	Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals	80
4.11	Computed Piecewise Constant Optimal Control, Extended by Zero for $t > 1800$, for Space Station Momentum Dumping Problem, Using N=30 on Five Subintervals.	81
4.12	Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals	82
4.13	Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals	83
4.14	Simulated External Torques for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals	84
4.15	Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals	86
4.16	Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals	87

4.17 Computed Piecewise Constant Optimal Control, Extended by Zero for $t > 1800$, for Space Station Momentum Dumping Problem, Using N=30 on Five Subintervals	88
4.18 Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals	89
4.19 Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals	90
4.20 Simulated External Torques for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals	91
4.21 Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Control Law, Using N=150	94
4.22 Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Control Law, Using N=150 . . .	95
4.23 Simulated, Optimal and Error Angular Momentum Time Derivative for Space Station Momentum Dumping Problem with Control Law, Using N=150	96
4.24 Computed Optimal Attitude Command Control, Extended by Zero for $t > 1800$, for the Space Station Momentum Dumping Problem, Using N=150	97
4.25 Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Control Law, Using N=150	98
4.26 Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Control Law, Using N=150 . . .	99

4.27 Simulated External Torques for Space Station Momentum Dumping Problem with Control Law, Using N=150	100
---	-----

List of Tables

4.1	ISS 12A Inertia Matrix [slugs-ft ²]	65
4.2	Attitude and Rate Corresponding to Principal Axis	67

Chapter 1

Introduction

Optimal control problems (OCPs) governed by ordinary differential equations arise in a wide range of applications. One particular field where these optimal control problems are abundant is the aerospace industry. Aerospace engineers have been solving optimal control problems for trajectory optimization, spacecraft attitude control, jet thruster control, missile guidance and many other applications for decades. Methods for obtaining these solutions are almost as copious as the applications themselves.

A traditional approach to solving OCPs entails forming the optimality conditions directly, using the calculus of variations and Pontryagin's maximum principle [42], and then solving the resulting equations to obtain the solution to the optimal control problem. This is known as the *indirect* approach for solving OCPs. The references [9, 23, 41, 42, 45, 46] present just a small sample of the work that discusses or applies indirect methods for the solution of optimal control problems. In rare cases the solution can be obtained analytically from these optimality conditions, but in general, approximation methods are used to solve the problem numerically. The optimality conditions of these problems generally take the form of differential algebraic equations (DAEs) or boundary value problems (BVPs). The approximate solution to the OCP can be obtained by using a BVP solver. Many such methods exist. Perhaps the most popular methods are multiple shooting and collocation. The reader is encouraged to consult [3] for more information on these and other numerical methods for solving BVPs.

Alternatively, one can discretize the governing ODEs and the integral terms in the objective functional or constraint functions and thereby replace the infinite dimen-

sional optimal control problem by a large nonlinear programming problem (NLP). This is known as the *direct or direct transcription* approach for solving OCPs. This approach is typically easier to use, especially for OCPs with state equality or inequality constraints. Direct methods have been used, e.g., in [6, 7, 16, 44].

This thesis focuses on a class of direct transcription methods in which the governing ODEs are discretized using pseudospectral collocation methods. Such methods have attracted attention [15, 14, 17, 18, 20, 43] because of their alleged superior approximation properties and, in the case of Legendre pseudospectral method, the availability of a so-called adjoint map or estimate. However, most of the existing work in this area is numerical with incomplete, informal discussions of mathematical properties of pseudospectral discretizations for optimal control problems.

The goals of this thesis are to improve the mathematical understanding of pseudospectral discretizations for optimal control problems and to apply these methods to the solution of optimal control problems with significance to the aerospace community. In particular, we provide a systematic derivation of adjoint estimates for all pseudospectral discretizations that use Gauss-Lobatto points and we present rigorous results on the error between the solution computed using pseudospectral discretizations and the exact solution of the underlying infinite dimensional OCP.

Adjoint estimates provide approximations to the adjoint variables (also known as costate variables) corresponding to the optimal solution of the OCP in terms of the Lagrange multipliers corresponding to the NLP derived using the direct transcription method. Such approximations are important for error analysis, mesh refinement strategies, and real-time optimization using the method of neighboring extrema. Among the few results on adjoint estimates are [17, 26, 27]. In the context of pseudospectral discretizations, only [17] have provided an adjoint estimation procedure for the particular case of Legendre pseudospectral discretizations. This thesis provides a systematic derivation of adjoint estimates for all Gauss-Lobatto pseudospectral discretizations, which as a special case includes the result of [17]. The work on ad-

joint estimation provides the foundation towards a rigorous convergence analysis that provides estimates for the error between the solution of the infinite dimensional optimal control problem and associated adjoint as well as the solution of the discretized optimal control problem and associated Lagrange multipliers. Such error estimates are not available in the existing literature. This thesis derives error estimates for linear-quadratic optimal control problems, and presents numerical results for both linear-quadratic and nonlinear optimal control example problems.

In the second part of this thesis, a class of pseudospectral direct transcription methods are applied to a series of optimal control problems derived from the International Space Station (ISS) momentum dumping problem. This is an attitude control problem where the attitude of the station is manipulated by a controller which uses control moment gyroscopes (CMGs). The issue here is that the CMGs have a maximum momentum threshold which cannot be exceeded. Doing so will result in loss of control of the vehicle. The goal is to find a control trajectory that will maneuver the attitude of the ISS from some initial state to some final state with minimal total momentum on the CMGs, obeying the system dynamics and never exceeding the momentum threshold along the way. What makes this problem difficult is the severe nonlinearity of the problem and the possible discrete nature of the controls. Related spacecraft control problems are discussed in [1, 5, 8, 13, 36, 38, 43, 45]. This thesis includes a study of the numerical solution to the ISS momentum dumping problem which demonstrates the utility of pseudospectral methods for the direct transcription of optimal control problems.

This thesis is organized as follows. Chapter 2 states the general form of the optimal control problems that will be considered, their corresponding optimality conditions and provides some examples problems that will be used throughout this thesis. Chapter 3 states the optimality conditions of the discretized OCP, describes how adjoint estimates are obtained and explores some of the consequences of using pseudospectral methods in the direct transcription of optimal control problems. The

application of the Legendre pseudospectral method to the space station momentum dumping problem is addressed in Chapter 4. Finally, Chapter 5 contains remarks, conclusions and suggestions for future work.

Chapter 2

Optimal Control Problems Governed by Ordinary Differential Equations

This chapter provides a description of the optimal control problems that are considered in this thesis. In addition, first order necessary optimality conditions are stated for the infinite dimensional problem and reformulations of optimal control problems are presented. This material is well known and will be used in subsequent chapters. Much of the material developed in Chapter 3 can be applied to OCPs that are of a more general form than (2.1). However (2.1) covers the applications considered in this thesis and is sufficient to describe the theoretical aspects of pseudospectral methods as used to transcribe optimal control problems.

2.1 Problem Statement and Necessary Optimality Conditions

In this thesis we consider optimal control problems of the following class

$$\min \quad m(y(t_f)) + \int_{t_0}^{t_f} \ell(y(t), u(t)) dt, \quad (2.1a)$$

s.t.

$$\frac{d}{dt} y(t) = f(y(t), u(t)), \quad (2.1b)$$

$$y(t_0) = \bar{y}_0, \quad (2.1c)$$

$$b(y(t_f)) = 0. \quad (2.1d)$$

Here $y : [t_0, t_f] \mapsto \mathbb{R}^{n_y}$ are the state variables and $u : [t_0, t_f] \mapsto \mathbb{R}^{n_u}$ are the control variables to be determined. The functions $m : \mathbb{R}^{n_y} \mapsto \mathbb{R}$, $\ell : \mathbb{R}^{n_y} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}$, $f : \mathbb{R}^{n_y} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}^{n_y}$ and $b : \mathbb{R}^{n_y} \mapsto \mathbb{R}^{n_b}$ as well as $\bar{y}_0 \in \mathbb{R}^{n_y}$ are given. The conditions

(2.1b) and (2.1c) are called the state equations. The problem (2.1) is said to be in Bolza form.

We seek solutions $u \in L_\infty^{n_u}[t_0, t_f]$ and $y \in W_{1,\infty}^{n_y}[t_0, t_f]$. The space $L_\infty[t_0, t_f]$ is the space of all Lebesgue-measurable functions with the property that their absolute value is essentially bounded on $[t_0, t_f]$ and $L_\infty^{n_u}[t_0, t_f] = (L_\infty[t_0, t_f])^{n_u}$, while

$$W_{1,\infty}^{n_y}[t_0, t_f] := \left\{ y : [t_0, t_f] \mapsto \mathbb{R}^{n_y} \mid y \text{ is absolutely continuous, } \frac{d}{dt}y \in L_\infty^{n_y}[t_0, t_f] \right\}.$$

These spaces are equipped with norms

$$\|u\|_{L_\infty^{n_u}[t_0, t_f]} = \operatorname{ess\ sup}_{t_0 \leq t \leq t_f} \|u(t)\|_2$$

and

$$\|y\|_{W_{1,\infty}^{n_y}[t_0, t_f]} = \max \left\{ \|y\|_{L_\infty^{n_y}[t_0, t_f]}, \left\| \frac{d}{dt}y \right\|_{L_\infty^{n_y}[t_0, t_f]} \right\}.$$

The necessary optimality conditions for (2.1) can be obtained using a generalization of the well-known Lagrange multiplier theorem. We need a constraint qualification to ensure the surjectivity of the linearized constraints (2.1b)–(2.1d) at the solution y_*, u_* . For (2.1) this can be guaranteed by assuming that the Jacobian $b_y(y_*(t_f))$ has full row rank and that the linearized state equations

$$\begin{aligned} \frac{d}{dt}y(t) &= f_y(y_*(t), u_*(t))^T y(t) + f_u(y_*(t), u_*(t))^T u(t), \\ &\quad \text{almost everywhere on } [t_0, t_f], \\ y(t_0) &= \bar{y}_0, \end{aligned} \tag{2.2}$$

are controllable. Controllable means that for every $y_f \in \mathbb{R}^{n_y}$ there exists a control $u \in L_\infty^{n_u}[t_0, t_f]$ such that the solution $y \in W_{1,\infty}^{n_y}[t_0, t_f]$ of (2.2) satisfies $y(t_f) = y_f$.

The following theorem is proven in [33, Sec. 5.4].

Theorem 2.1 Let the optimal control problem (2.1) be given. Let the functions m, ℓ, f and b be continuously partially differentiable. Let $u_* \in L_\infty^{n_u}[t_0, t_f]$ be an optimal control and let $y_* \in W_{1,\infty}^{n_y}[t_0, t_f]$ be the resulting state. Let the matrix $b_y(y_*(t_f))$ have full row rank and let the linearized

system (2.2) be controllable. Then there exists a function $p_* \in W_{1,\infty}^{n_y}[t_0, t_f]$ and vectors $(q_0)_* \in \mathbb{R}^{n_y}$ and $(q_f)_* \in \mathbb{R}^{n_b}$ such that the boundary value problem

$$\begin{aligned} \frac{d}{dt}y_*(t) &= f(y_*(t), u_*(t)), \\ y_*(t_0) &= \bar{y}_0, \end{aligned} \tag{2.3a}$$

$$b(y_*(t_f)) = 0,$$

$$\begin{aligned} \frac{d}{dt}p_*(t) &= -f_y(y_*(t), u_*(t))^T p_*(t) - \ell_y(y_*(t), u_*(t)), \\ &\quad (\text{adjoint equation}) \end{aligned} \tag{2.3b}$$

$$p_*(t_0) = -(q_0)_*,$$

$$\begin{aligned} p_*(t_f) &= m_y(y_*(t_f)) + b_y(y_*(t_f))^T (q_f)_*, \\ &\quad (\text{transversality conditions}) \end{aligned} \tag{2.3c}$$

$$\begin{aligned} 0 &= f_u(y_*(t), u_*(t))^T p_*(t) + \ell_u(y_*(t), u_*(t)), \\ &\quad (\text{local Pontryagin maximum principle}) \end{aligned} \tag{2.3d}$$

are satisfied almost everywhere on $[t_0, t_f]$.

The conditions (2.3b),(2.3c) and (2.3d) are obtained by computing the Fréchet derivatives of the Lagrangian function

$$\begin{aligned} \mathcal{L}(y, u, p, q_0, q_f) &= m(y(t_f)) + b(y(t_f))^T q_f + (y(t_0) - \bar{y}_0)^T q_0 \\ &\quad + \int_{t_0}^{t_f} \ell(y(t), u(t)) + p(t)^T [f(y(t), u(t)) - \frac{d}{dt}y(t)] dt, \end{aligned} \tag{2.4}$$

with respect to y and u and setting these Fréchet derivatives to zero.

2.2 Bolza and Mayer Forms of the Optimal Control Problem

For most of this thesis, optimal control problems of the following Mayer form are considered,

$$\min \quad m(y(t_f)), \quad (2.5a)$$

s.t.

$$\frac{d}{dt}y(t) = f(y(t), u(t)), \quad (2.5b)$$

$$y(t_0) = \bar{y}_0, \quad (2.5c)$$

$$b(y(t_f)) = 0. \quad (2.5d)$$

This is no restriction, since every problem (2.1) in Bolza form can be converted into an equivalent problem in Mayer form. This will be discussed shortly.

The optimality conditions for the Mayer form optimal control problem can be obtained as an application of Theorem 2.1. They are stated here for later reference. They consist of the boundary value problem

$$\begin{aligned} \frac{d}{dt}y(t) &= f(y(t), u(t)), \\ y(t_0) &= \bar{y}_0, \\ b(y(t_f)) &= 0, \end{aligned} \quad (2.6a)$$

the adjoint equation

$$\frac{d}{dt}p(t) = -f_y(y(t), u(t))^T p(t), \quad (2.6b)$$

with transversality conditions

$$\begin{aligned} p(t_0) &= -q_0, \\ p(t_f) &= m_y(y(t_f)) + b_y(y(t_f))^T q_f, \end{aligned} \quad (2.6c)$$

and the gradient equation

$$f_u(y(t), u(t))^T p(t) = 0. \quad (2.6d)$$

One can transform a Bolza problem into a Mayer problem by moving the integral term in (2.1) into the differential equation by defining an auxiliary variable

$$z(t) = \int_{t_0}^t \ell(y(\tau), u(\tau)) d\tau. \quad (2.7)$$

If (2.7) is differentiated with respect to t the following initial value problem (IVP) for z arises

$$\begin{aligned} \frac{d}{dt} z(t) &= \ell(y(t), u(t)), \\ z(t_0) &= 0. \end{aligned} \quad (2.8)$$

The IVP (2.7) could then be inserted into the differential equation in (2.5b) along with y . This leads to the following optimal control problem (2.9) in Mayer form, which is equivalent to (2.1).

$$\min \quad m(y(t_f)) + z(t_f), \quad (2.9a)$$

s.t.

$$\frac{d}{dt} y(t) = f(y(t), u(t)), \quad (2.9b)$$

$$\frac{d}{dt} z(t) = \ell(y(t), u(t)), \quad (2.9c)$$

$$y(t_0) = \bar{y}_0, \quad (2.9d)$$

$$z(t_0) = 0, \quad (2.9e)$$

$$b(y(t_f)) = 0. \quad (2.9f)$$

Application of Theorem 2.1 to (2.9) leads to the following necessary optimality conditions. There exist adjoint variables p associated with (2.9b) and r associated with (2.9c) as well as multipliers q_0, q_f associated with (2.9d), (2.9f) and multipliers s_0 associated with (2.9e) such that the constraints

$$\begin{aligned} \frac{d}{dt} y(t) &= f(y(t), u(t)), \\ y(t_0) &= \bar{y}_0, \\ b(y(t_f)) &= 0, \end{aligned} \quad (2.10a)$$

and

$$\begin{aligned}\frac{d}{dt}z(t) &= \ell(y(t), u(t)), \\ z(t_0) &= 0,\end{aligned}\tag{2.10b}$$

are satisfied, the adjoint equation

$$\frac{d}{dt}p(t) = -f_y(y(t), u(t))^T p(t) - \ell_y(y(t), u(t))r(t),\tag{2.10c}$$

the auxiliary adjoint equation

$$\frac{d}{dt}r(t) = 0,\tag{2.10d}$$

the transversality conditions

$$\begin{aligned}p(t_0) &= -q_0, \\ p(t_f) &= m_y(y(t_f)) + b_y(y(t_f))^T q_f,\end{aligned}\tag{2.10e}$$

the auxiliary transversality conditions

$$\begin{aligned}r(t_0) &= -s_0, \\ r(t_f) &= 1,\end{aligned}\tag{2.10f}$$

are satisfied, and the gradient equation

$$f_u(y(t), u(t))^T p(t) + \ell_u(y(t), u(t))r(t) = 0,\tag{2.10g}$$

holds.

Note that (2.10d) and (2.10f) imply $r(t) = 1$. The necessary optimality conditions for (2.1) and (2.9) are equivalent.

The conversion of the problem (2.1) in Bolza form into a problem (2.9) in Mayer form is also important from a numerical point of view. For an efficient numerical solution it is important that the discretization of the state equation (2.1b) is consistent with the discretization of the integral in (2.1a). This is not straightforward for many high order discretization methods. This difficulty is avoided when (2.1) is transformed into (2.9), since only a system of differential equations has to be discretized. The

discretization of the z -component of this system implicitly defines a discretization of the integral term in (2.1a) that is consistent with the discretization of (2.1b). For additional discussions see, e.g., [7].

2.3 Prototypical Examples

Throughout this thesis two example problems are used to demonstrate various properties associated with solving optimal control problems using a pseudospectral direct transcription method. These problems are stated here so that they may be referred to elsewhere.

Example 2.2 (*Linear-Quadratic Optimal Control Problem*) This problem was adapted from [27]. Consider the following linear-quadratic optimal control problem

$$\begin{aligned} \min \quad & \int_0^1 y(t)^2 + \frac{1}{2}u(t)^2 dt, \\ \text{s.t.} \quad & \frac{d}{dt}y(t) = \frac{1}{2}y(t) + u(t), \quad t \in [0, 1], \\ & y(0) = 1. \end{aligned} \tag{2.11}$$

Simple evaluation of the necessary conditions (2.3) leads to the following exact solution for the state

$$y_*(t) = \frac{2e^{3t} + e^3}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1], \tag{2.12a}$$

the control

$$u_*(t) = \frac{2(e^{3t} - e^3)}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1], \tag{2.12b}$$

and the adjoint

$$p_*(t) = -\frac{2(e^{3t} - e^3)}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1]. \tag{2.12c}$$

The problem (2.11) can be equivalently written as a problem in Mayer from:

$$\min \quad y_2(1),$$

s.t.

$$\begin{aligned} \frac{d}{dt}y_1(t) &= \frac{1}{2}y_1(t) + u(t), \quad t \in [0, 1], \\ \frac{d}{dt}y_2(t) &= y_1(t)^2 + \frac{1}{2}u(t)^2, \quad t \in [0, 1], \\ y_1(0) &= 1, \\ y_2(0) &= 0. \end{aligned} \tag{2.13}$$

Evaluation of the necessary conditions (2.6) leads to the following exact solution for the state

$$(y_1)_*(t) = \frac{2e^{3t} + e^3}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1], \tag{2.14a}$$

the auxiliary state

$$(y_2)_*(t) = \frac{e^{-3t}(2e^{6t} + (e^6 - 2)e^{3t}) - e^6}{(2 + e^3)^2}, \quad t \in [0, 1], \tag{2.14b}$$

the control

$$u_*(t) = \frac{2(e^{3t} - e^3)}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1], \tag{2.14c}$$

the adjoint

$$(p_1)_*(t) = -\frac{2(e^{3t} - e^3)}{e^{3t/2}(2 + e^3)}, \quad t \in [0, 1], \tag{2.14d}$$

and the auxiliary adjoint

$$(p_2)_*(t) = 1, \quad t \in [0, 1]. \tag{2.14e}$$

Example 2.3 (Orbit Transfer Optimal Control Problem) This example is adapted from [8]. This problem is frequently used in the context of pseudospectral direct transcription methods for optimal control problems, see [15, 14, 17]. Consider the following orbit transfer optimal control problem of finding an optimal trajectory and thrust steering vector to

transfer a spacecraft from an initial orbit to a final orbit in a fixed amount of time. This problem is stated as

$$\begin{aligned}
 \min \quad & -\frac{1}{2}y_2(50)^2 - \frac{1}{2}y_3(50)^2 + y_1(50)^{-1}, \\
 \text{s.t.} \quad & \\
 \frac{d}{dt}y_1(t) &= y_2(t), \quad t \in [0, 50], \\
 \frac{d}{dt}y_2(t) &= \frac{y_3(t)^2}{y_1(t)} - \frac{1}{y_1(t)^2} + 0.01 \sin(u(t)), \quad t \in [0, 50], \\
 \frac{d}{dt}y_3(t) &= -\frac{y_2(t)y_3(t)}{y_1(t)} + 0.01 \cos(u(t)), \quad t \in [0, 50], \\
 y_1(0) &= 1.1, \\
 y_2(0) &= 0, \\
 y_3(0) &= 1/\sqrt{1.1},
 \end{aligned} \tag{2.15}$$

where y_1 is the state which describes radial distance, y_2 is the state which describes the radial component of velocity, y_3 is the state which describes the tangential component of velocity, and u is the controllable thrust steering angle. It should be noted that this problem has no analytical solution, however numerical solutions to this problem can be found in [15, 14, 17].

Chapter 3

Direct Transcription of Optimal Control Problems

In this chapter we describe and analyze pseudospectral collocation discretization of the optimal control problem (2.5). Such direct transcription methods, especially Chebyshev collocation and Legendre collocation methods have received significant attention recently [14, 16, 17, 19, 29, 30]. One reason for this is the alleged fast convergence of the solutions of discretized optimal control problem to the solution of the underlying infinite dimensional control problem. Another reason for the large interest in direct transcription methods based on Legendre collocation is the availability of an adjoint estimate that relates the Lagrange multipliers of the discretized problem to the adjoint variables p evaluated at collocation points.

While there is numerical evidence that shows fast convergence of the solutions of discretized optimal control problem to the solution of the underlying infinite dimensional control problem, the theoretical foundation is largely missing. The papers [14, 17, 19] cite estimates in [10, 25] for errors between a function (its derivatives) and its interpolant (derivatives of its interpolant). Such a result may be used to establish consistency results as one step in the argument that pseudospectral collocation are fast converging schemes for the solution of the dynamics for a given fixed control. But these results are not sufficient to establish convergence of the solution of the discretized optimal control problem to the solution of the underlying infinite dimensional control problem. In fact, [27] contains simple examples using Runge-Kutta discretizations of optimal control problems, which show that the solution of the discretized optimal control problem may converge to the solution of the underly-

ing infinite dimensional control problem with a slower rate than one might expect, or may not converge at all.

The goal of this section is to obtain a better understanding of pseudospectral collocation discretizations of the optimal control problem (2.5), in particular to obtain a better theoretical foundation for their observed fast convergence. An important step towards this goal is obtaining estimates for the error between the true solution to the OCP and the computed solution to the OCP. To obtain these error estimates, the derivation of an adjoint estimate that relates the Lagrange multipliers of the discretized problem to the adjoint variables p evaluated at collocation points is necessary. Such adjoint estimates are also important because the adjoint variables p are used, e.g., in the method of neighboring extrema for real-time optimal control. Despite the theoretical and practical importance of adjoint estimates, there are few results for high order discretizations. We will derive, in a systematic way, adjoint estimates for a large class of pseudospectral collocation discretizations of the optimal control problem (2.5). The adjoint estimation procedure of [17] is a special case of our treatment. We also derive some important estimates for the error between the solution of the discretized optimal control problem and the solution of the underlying infinite dimensional control problem for linear quadratic problems. For a linear quadratic optimal control problem we investigate stability results numerically. Finally, we comment on the discretization of optimal control problems in Bolza form using pseudospectral collocation discretizations.

Since the reference interval $[t_0, t_f]$ can always be transformed to the standard interval $[-1, 1]$ using the change of variables

$$t \mapsto -1 + 2(t - t_0)/(t_f - t_0),$$

it is assumed that

$$[t_0, t_f] = [-1, 1]$$

throughout this chapter, except for Section 3.6.

3.1 Pseudospectral Discretization of the Optimal Control Problem

For the numerical solution of the optimal control problem we discretize (2.5) using a pseudospectral collocation method with collocation points

$$c_0 = -1, \quad c_1, \dots, c_{N-1} \in (-1, 1), \quad c_N = 1.$$

The state y is approximated by the polynomial

$$y^N(t) = \sum_{j=0}^N y_j \psi_j(t), \quad (3.1)$$

where

$$\psi_j(t) = \prod_{\substack{l=1 \\ l \neq j}}^N \frac{t - c_l}{c_j - c_l}, \quad (3.2)$$

$j = 0, \dots, N$, is the j th Lagrange polynomial. Clearly,

$$y^N(c_j) = y_j, \quad j = 0, \dots, N.$$

Furthermore,

$$\frac{d}{dt} y^N(c_j) = \sum_{k=0}^N y_k \frac{d}{dt} \psi_k(c_j)$$

and

$$\begin{pmatrix} \frac{d}{dt} y^N(c_0) \\ \vdots \\ \frac{d}{dt} y^N(c_N) \end{pmatrix} = \mathbf{D} \begin{pmatrix} y_0 \\ \vdots \\ y_N \end{pmatrix}, \quad (3.3)$$

where $\mathbf{D} = D \otimes I_{n_y} \in \mathbb{R}^{n_y(N+1) \times n_y(N+1)}$ is the Kronecker product between the so-called differentiation matrix $D \in \mathbb{R}^{(N+1) \times (N+1)}$ with entries

$$D_{jk} = \frac{d}{dt} \psi_k(c_j), \quad j, k = 1, \dots, N+1, \quad (3.4)$$

and the $n_y \times n_y$ identity matrix I_{n_y} . Recall that the Kronecker product of two matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{k \times l}$ is defined as

$$A \otimes B = \begin{pmatrix} A_{11}B & \cdots & A_{1n}B \\ \vdots & & \vdots \\ A_{m1}B & \cdots & A_{mn}B \end{pmatrix} \in \mathbb{R}^{mk \times nl}.$$

To discretize the state equation (2.5b), we substitute y by y^N given by (3.1) and require that the ODE (2.5b) holds at the collocation points c_0, \dots, c_N . Furthermore, we insert (3.1) into the objective function (2.5a), the initial condition (2.5c), and the final time condition (2.5d). This leads to the following pseudospectral collocation discretization of (2.5).

$$\min \quad m(y_N), \quad (3.5a)$$

s.t.

$$\mathbf{D} \begin{pmatrix} y_0 \\ \vdots \\ y_N \end{pmatrix} = \begin{pmatrix} f(y_0, u_0) \\ \vdots \\ f(y_N, u_N) \end{pmatrix}, \quad (3.5b)$$

$$y_0 = \bar{y}_0, \quad (3.5c)$$

$$b(y_N) = 0. \quad (3.5d)$$

It is important to note that, following [14, 17, 19] and others, we include a collocation condition at c_0 as well as the initial condition as constraints in (3.5). This is different from other direct transcription methods based on collocation, where only the collocation conditions at c_1, \dots, c_N as well as the initial condition are included as constraints (see e.g., [4, 7, 44]). This also seems different from the way pseudospectral methods are used to discretize boundary value problems, where one also eliminates collocation conditions at c_0, c_N , depending on the type of boundary conditions specified (see, e.g., [21, 48, 49]).

With the inclusion of the collocation condition at c_0 it is, in general, not possible to solve the $n_y(N+2)$ discretized state equations (3.5b), (3.5c) for the $n_y(N+1)$ states y_0, \dots, y_N given controls u_0, \dots, u_N , even if the infinite dimensional state equation (2.5b), (2.5c) has a unique solution y for given control u . This is quite different from the collocation discretizations [4, 7, 44], where the discretized state equation consists of $n_y(N+1)$ equations and where, under suitable assumptions, the discretized state equation has a unique solution y_0, \dots, y_N , given controls u_0, \dots, u_N . Hence the

discretization (3.5b), (3.5c) of the state equation (2.5b), (2.5c) only makes sense in the context of optimal control, but not for simulations.

The choice of including the collocation condition at c_0 leads to nice adjoint estimation properties, which will be discussed in Section 3.3. On the other hand, it is not obvious that the linearization of the constraints (3.5b)–(3.5d) has full row rank, even if the constraint qualification in Theorem 2.1 holds for the infinite dimensional problem. It is possible to check this condition a priori for problems with linear constraints, but for problems with nonlinear constraints, it is typical to assume that this constraint qualification holds and then verify it after computing the estimated solution, see [47]. In the next example we examine the numerical rank of the linearized constraints for the discretization of the two problems stated in Examples 2.13 and 2.15.

Example 3.1 In this example, the numerical rank of the constraint Jacobian in (3.5), evaluated at the computed solutions and using Legendre pseudospectral collocation, are investigated for the problems in Examples 2.13 and 2.15. The numerical rank is investigated by inspecting the ratio between smallest singular value and largest singular value of the constraint Jacobian. Figure 3.1 depicts the ratio of the minimum singular value of the constraint Jacobian divided by the maximum singular of the constraint Jacobian value for different N . While for fixed N the constraint Jacobians have full rank, Figure 3.1 indicates that one should expect numerical rank deficiency for large N .

We conclude this section by stating a few facts about the two pseudospectral collocation methods that have been used for the direct transcription of optimal control problems [14, 17, 19]. Details about the computation of collocation points, corresponding differentiation matrices and quadrature weights may be found, e.g., in [10, 21, 34, 49].

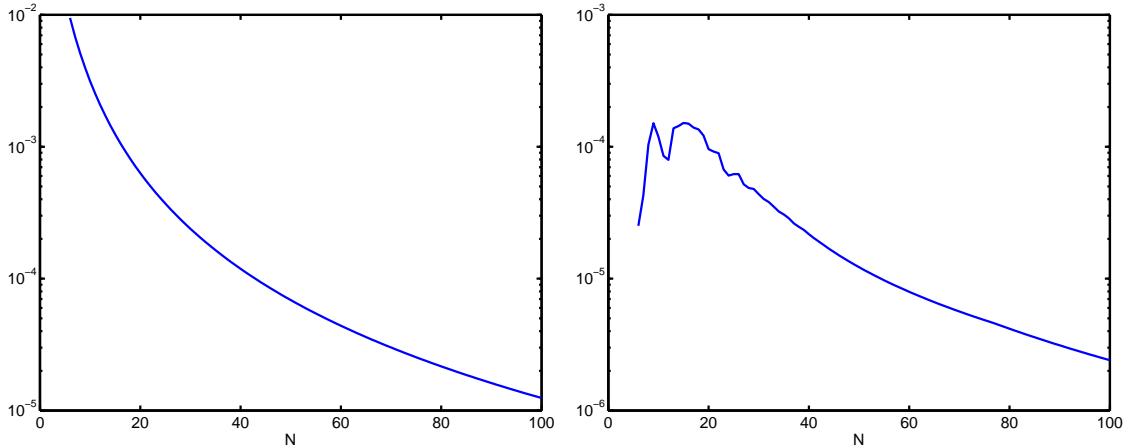


Figure 3.1: Ratio Between Minimum Singular Value and Maximum Singular Value of the Constraint Jacobian at the Solution for Various N . - Left: Linear-Quadratic OCP in Mayer Form - Right: Orbit Transfer OCP

Example 3.2 (Chebyshev-Gauss-Lobatto Collocation) The Chebyshev-Gauss-Lobatto collocation points are

$$c_j = -\cos\left(\frac{j\pi}{N}\right), \quad j = 0, \dots, N. \quad (3.6)$$

The c_j 's are the extrema of the Chebyshev polynomials $T_N(t)$ in $[-1, 1]$. The c_j 's are also the roots of $(1-t^2)\frac{d}{dt}T_N(t)$ in $[-1, 1]$. The corresponding differentiation matrix is given by

$$D_{jk} = \begin{cases} -\frac{\xi_k(-1)^{j+k}}{\xi_j(c_k - c_j)} & j \neq k, \\ -\frac{2N^2+1}{6} & j = k = 0, \\ \frac{2N^2+1}{6} & j = k = N, \\ \frac{1}{2} \frac{c_k}{(1-c_k^2)} & \text{otherwise,} \end{cases} \quad (3.7)$$

where $\xi_0 = \xi_N = 2$ and $\xi_1 = \dots = \xi_{N-1} = 1$.

Note that the points $c_j = -\cos(j\pi/N)$ are defined so that $-1 = c_0 < c_1 < \dots < c_{N-1} < c_p = 1$. In the literature, one often finds the definition

$c_j = +\cos(j\pi/p)$. In the latter case the signs in the differentiation matrix given in (3.7) need to be reversed.

With the weighting function

$$g(t) = 1/\sqrt{1-t^2},$$

and the quadrature weights

$$w_j = \int_{-1}^1 g(t)\psi_j(t)dt = \begin{cases} \pi/N & 1 \leq j \leq N-1, \\ \pi/(2N) & j=0, N, \end{cases} \quad (3.8)$$

the quadrature formula

$$\int_{-1}^1 g(t)h(t)dt \approx \sum_{j=0}^N w_j h(c_j), \quad (3.9)$$

is exact of degree $2N-1$, i.e.,

$$\int_{-1}^1 g(t)h(t)dt = \sum_{j=0}^N w_j h(c_j) \quad \forall h \in \mathcal{P}_{2N-1}([-1, 1]).$$

Example 3.3 (Legendre-Gauss-Lobatto Collocation) The Legendre-Gauss-Lobatto collocation points c_j , $j = 0, \dots, N$, are the roots of $(1-t^2)\frac{d}{dt}L_N(t)$, where $L_N(t)$ is the Legendre polynomial of degree N . The differentiation matrix for the Legendre-Gauss-Lobatto collocation is given by

$$D_{jk} = \begin{cases} \frac{L_N(c_j)}{L_N(c_k)} \frac{1}{c_j - c_k} & j \neq k, \\ \frac{-N(N+1)}{4} & j = k = 0, \\ \frac{N(N+1)}{4} & j = k = N, \\ 0 & \text{otherwise.} \end{cases} \quad (3.10)$$

The weighting function is

$$g(t) = 1,$$

and the quadrature weights are

$$w_j = \int_{-1}^1 \psi_j(t)dt = \frac{2}{N(N+1)} \frac{1}{[L_N(c_j)]^2}. \quad (3.11)$$

The quadrature formula (3.9) is exact of degree $2N-1$.

3.2 Optimality Conditions for the Discretized Optimal Control Problem

The Lagrangian corresponding to (3.5) is given by

$$\begin{aligned} L(\mathbf{y}, \mathbf{u}, \tilde{\boldsymbol{\lambda}}, \mu_0, \mu_N) &= m(y_N) \\ &\quad + \sum_{j=0}^N \tilde{\lambda}_j^T \left[f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right] \\ &\quad + \mu_0^T (y_0 - \bar{y}_0) + \mu_N^T b(y_N), \end{aligned} \quad (3.12)$$

where

$$\mathbf{y} = (y_0^T, \dots, y_N^T)^T,$$

$$\mathbf{u} = (u_0^T, \dots, u_N^T)^T,$$

and

$$\tilde{\boldsymbol{\lambda}} = (\tilde{\lambda}_0^T, \dots, \tilde{\lambda}_N^T)^T.$$

Differentiating the Lagrangian (3.12) with respect to the y_j 's and setting the derivatives to zero gives the *discrete adjoint equations*

$$\begin{aligned} f_y(y_0, u_0)^T \tilde{\lambda}_0 - \sum_{k=0}^N D_{k,0} \tilde{\lambda}_k &= -\mu_0, \\ f_y(y_j, u_j)^T \tilde{\lambda}_j - \sum_{k=0}^N D_{k,j} \tilde{\lambda}_k &= 0, \quad j = 1, \dots, N-1, \\ f_y(y_N, u_N)^T \tilde{\lambda}_N - \sum_{k=0}^N D_{k,N} \tilde{\lambda}_k &= -b_y(y_N)^T \mu_N - m_y(y_N). \end{aligned} \quad (3.13a)$$

Differentiating the Lagrangian (3.12) with respect to the u_j 's and setting the derivatives to zero gives the *discrete gradient equations*

$$f_u(y_j, u_j)^T \tilde{\lambda}_j = 0, \quad j = 0, \dots, N. \quad (3.13b)$$

For completeness we also state the *discrete state constraints*

$$\begin{aligned} f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k &= 0, \quad j = 0, \dots, N, \\ y_0 - \bar{y}_0 &= 0, \\ b(y_N) &= 0, \end{aligned} \quad (3.13c)$$

which are obtained by differentiating the Lagrangian (3.12) with respect to $\tilde{\lambda}_j$, $j = 0, \dots, N$, μ_0 and μ_N and setting the derivatives to zero.

Note that the discrete adjoint equations (3.13a) consist only of $n_y(N+1)$ equations for the $n_y(N+2) + n_b$ Lagrange multipliers. Therefore the discrete adjoint equations alone do not specify the Lagrange multipliers, given u_j, y_j , $j = 0, \dots, N$. This is a consequence of the fact that we include both the collocation condition at c_0 and the initial conditions as constraints into our discretized optimal control problem (3.5) and therefore have $n_y(N+2)$ discrete state equations (3.5b), (3.5c) for $n_y(N+1)$ discrete state variables y_j , $j = 0, \dots, N$.

3.3 Pseudospectral Method Adjoint Estimation Properties

As stated earlier, a goal of this thesis is to derive an estimate for the error between the solution of the optimal control problem (2.5) and the solution of the discretized problem (3.5). One step towards this goal is to identify a suitable adjoint estimation procedure, i.e., to compare the Lagrange multipliers $\tilde{\lambda}_j$ and the adjoints p evaluated at the collocation points c_j . Such adjoint estimates are not only useful for the error estimate described above. For many applications it is important to know an approximation of the adjoint variables p of the infinite dimensional control problem (2.5). Adjoint information is, for example, important to adaptive mesh refinement for optimal control problems. It is also important in real-time optimization of optimal control problems using neighboring extrema [12, 35, 52]. Despite its importance, there are few results about adjoint estimation. Adjoint estimates for optimal control problems discretized using Runge-Kutta methods are discussed in [27]. An adjoint estimation procedure for the Legendre-Gauss-Lobatto pseudospectral discretization of optimal control problems is presented in [17]. In this section, we will derive adjoint estimates for a broad class of pseudospectral discretization of optimal control problems. Our class of adjoint estimates includes those presented in [17] as a special case and is derived in a systematic way.

3.3.1 Integration by Parts Approach

Let c_0, \dots, c_N be the collocation points, let $g : [-1, 1] \rightarrow \mathbb{R}$ be a positive weighting function and let w_0, \dots, w_N be positive weights such that the quadrature formula

$$\int_{-1}^1 g(t)h(t)dt \approx \sum_{j=0}^N w_j h(c_j), \quad (3.14)$$

is exact of degree $2N - 1$, i.e.,

$$\int_{-1}^1 g(t)h(t)dt = \sum_{j=0}^N w_j h(c_j) \quad \forall h \in \mathcal{P}_{2N-1}([-1, 1]). \quad (3.15)$$

Instead of (3.12) consider the weighted Lagrangian

$$\begin{aligned} L_w(\mathbf{y}, \mathbf{u}, \boldsymbol{\lambda}, \mu_0, \mu_N) &= m(y_N) + (y_0 - \bar{y}_0)^T \mu_0 + b(y_N)^T \mu_N \\ &\quad + \sum_{j=0}^N w_j \lambda_j^T \left[f(y_j, u_j) - \sum_{k=0}^N D_{jk} y_k \right]. \end{aligned} \quad (3.16)$$

Clearly, if $\tilde{\lambda}_j = w_j \lambda_j$, then $L(\mathbf{y}, \mathbf{u}, \tilde{\boldsymbol{\lambda}}, \mu_0, \mu_N) = L_w(\mathbf{y}, \mathbf{u}, \boldsymbol{\lambda}, \mu_0, \mu_N)$. We define the polynomials

$$\begin{aligned} y^N(t) &= \sum_{k=0}^N y_k \psi_k(t), \\ u^N(t) &= \sum_{k=0}^N u_k \psi_k(t), \end{aligned}$$

and

$$\lambda^N(t) = \sum_{k=0}^N \lambda_k \psi_k(t).$$

Note that since

$$\frac{d}{dt} y^N(t) = \sum_{k=0}^N y_k \frac{d}{dt} \psi_k(t) \in \mathcal{P}_{N-1}([-1, 1]),$$

and $\lambda^N(t) \in \mathcal{P}_N([-1, 1])$, the equation (3.15) implies that

$$\begin{aligned} \sum_{j=0}^N w_j \lambda_j^T \sum_{k=0}^N D_{jk} y_k &= \sum_{j=0}^N w_j (\lambda^N(c_j))^T \frac{d}{dt} y^N(c_j) \\ &= \int_{-1}^1 g(t) (\lambda^N(t))^T \frac{d}{dt} y^N(t) dt. \end{aligned}$$

Furthermore, (3.14) yields the approximation

$$\begin{aligned} \sum_{j=0}^N w_j \lambda_j^T f(y_j, u_j) &= \sum_{j=0}^N w_j (\lambda^N(c_j))^T f(y^N(c_j), u^N(c_j)) \\ &\approx \int_{-1}^1 g(t) (\lambda^N(t))^T f(y^N(t), u^N(t)) dt. \end{aligned}$$

Hence, one may interpret

$$\begin{aligned} L_w(\mathbf{y}, \mathbf{u}, \boldsymbol{\lambda}, \mu_0, \mu_N) &\approx m(y^N(1)) + (y^N(-1) - \bar{y}_0)^T \mu_0 + b(y^N(1))^T \mu_N \\ &\quad + \int_{-1}^1 g(t) (\lambda^N(t))^T \left[\frac{d}{dt} y^N(t) - f(y^N(t), u^N(t)) \right] dt. \end{aligned} \quad (3.17)$$

This suggests the following relation between the Lagrange multipliers $\tilde{\lambda}_j$ and the weighted Lagrange multipliers λ_j of the direct transcription and the adjoint variables p

$$\frac{1}{w_j} \tilde{\lambda}_j = \lambda_j \approx \frac{p(c_j)}{g(c_j)}. \quad (3.18)$$

The relation (3.18) means that $\tilde{\lambda}_j/w_j = \lambda_j$ should be identified with $p(c_j)/g(c_j)$. In general $\tilde{\lambda}_j/w_j = \lambda_j$ is not identical to $p(c_j)/g(c_j)$. However, in Section 3.4 we will give conditions that guarantee

$$\frac{1}{w_j} \tilde{\lambda}_j \rightarrow \frac{p(c_j)}{g(c_j)},$$

as $N \rightarrow \infty$.

The necessary optimality conditions associated with (3.16), obtained by differentiating the weighted Lagrangian with respect to each variable and setting the result to zero are given by the *weighted-discrete adjoint equations*

$$\begin{aligned} f_y(y_0, u_0)^T w_0 \lambda_0 - \sum_{k=0}^N D_{k0} w_k \lambda_k &= -\mu_0, \\ f_y(y_j, u_j)^T w_j \lambda_j - \sum_{k=0}^N D_{kj} w_k \lambda_k &= 0, \quad j = 1, \dots, N-1, \\ f_y(y_N, u_N)^T w_N \lambda_N - \sum_{k=0}^N D_{kN} w_k \lambda_k &= -b_y(y_N)^T \mu_N - m_y(y_N), \end{aligned} \quad (3.19a)$$

by the *weighted-discrete gradient equations*

$$f_u(y_j, u_j)^T w_j \lambda_j = 0, \quad j = 0, \dots, N, \quad (3.19b)$$

and by the *weighted-discrete boundary value problem*

$$\begin{aligned} w_j \left(f(y_j, u_j) - \sum_{k=0}^N D_{jk} y_k \right) &= 0, \quad j = 0, \dots, N, \\ y_0 - \bar{y}_0 &= 0, \\ b(y_N) &= 0. \end{aligned} \tag{3.19c}$$

Note that these optimality conditions (3.19) can also be obtained from (3.13) by substituting $\tilde{\lambda}_j = w_j \lambda_j$ in (3.13).

The following notation will be useful. For a given function $f : [-1, 1] \rightarrow \mathbb{R}$ we define its interpolating polynomial

$$P_N(f)(t) = \sum_{i=0}^N f(c_i) \psi_i(t). \tag{3.20}$$

Note that

$$P_N(f)(c_i) = f(c_i), \quad i = 0, \dots, N,$$

and

$$\frac{d}{dt} P_N(f)(c_i) = \sum_{j=0}^N D_{ij} f(c_j), \quad i = 0, \dots, N. \tag{3.21}$$

The following lemma provides a discrete integration by parts formula.

Lemma 3.1 Let z_0, \dots, z_N be arbitrary and define $z^N(t) = \sum_{i=0}^N z_i \psi_i(t)$.

For any continuously differentiable function $p : [-1, 1] \rightarrow \mathbb{R}$ the equation

$$\begin{aligned} &\sum_{i=0}^N \frac{w_i}{g(c_i)} p(c_i) \sum_{j=0}^N D_{ij} z_j \\ &= p(c_N) z_N - p(c_0) z_0 - \sum_{i=0}^N \frac{w_i}{g(c_i)} z_i \sum_{j=0}^N D_{ij} p(c_j) \\ &\quad + \int_{-1}^1 g(t) \left(P_N \left(\frac{p}{g} \right)(t) - \frac{p(t)}{g(t)} \right) \frac{d}{dt} z^N(t) dt \\ &\quad + \int_{-1}^1 g(t) \left(P_N \left(\frac{z^N}{g} \right)(t) - \frac{z^N(t)}{g(t)} \right) \frac{d}{dt} P_N(p)(t) dt \\ &\quad + \int_{-1}^1 g(t) \frac{z^N(t)}{g(t)} \frac{d}{dt} (P_N(p)(t) - p(t)) dt \end{aligned} \tag{3.22}$$

holds. Furthermore,

$$\begin{aligned} \sum_{i=0}^N \frac{w_i}{g(c_i)} p(c_i) D_{i,k} &= p(c_N) \delta_{kN} - p(c_0) \delta_{k0} \\ &\quad - \frac{w_k}{g(c_k)} \sum_{i=0}^N D_{ki} p(c_i) + \epsilon_k(N, p, g), \end{aligned} \quad (3.23)$$

$k = 0, \dots, N$, where δ_{jk} is the Kronecker delta and

$$\begin{aligned} \epsilon_k(N, p, g) &= \int_{-1}^1 g(t) \left(P_N\left(\frac{p}{g}\right)(t) - \frac{p(t)}{g(t)} \right) \frac{d}{dt} \psi_k(t) dt \\ &\quad + \int_{-1}^1 g(t) \left(P_N\left(\frac{\psi_k}{g}\right)(t) - \frac{\psi_k(t)}{g(t)} \right) \frac{d}{dt} P_N(p)(t) dt \\ &\quad + \int_{-1}^1 g(t) \frac{\psi_k(t)}{g(t)} \frac{d}{dt} (P_N(p)(t) - p(t)) dt. \end{aligned} \quad (3.24)$$

Proof The definition of P_N and equation (3.15) imply

$$\begin{aligned} \sum_{i=0}^N w_i \frac{p(c_i)}{g(c_i)} \sum_{j=0}^N D_{ij} z_j &= \int_{-1}^1 g(t) P_N\left(\frac{p}{g}\right)(t) \frac{d}{dt} z^N(t) dt \\ &= \int_{-1}^1 p(t) \frac{d}{dt} z^N(t) dt + \int_{-1}^1 g(t) \left(P_N\left(\frac{p}{g}\right)(t) - \frac{p(t)}{g(t)} \right) \frac{d}{dt} z^N(t) dt \\ &= p(c_N) z_N - p(c_0) z_0 - \int_{-1}^1 \frac{d}{dt} p(t) z^N(t) dt \\ &\quad + \int_{-1}^1 g(t) \left(P_N\left(\frac{p}{g}\right)(t) - \frac{p(t)}{g(t)} \right) \frac{d}{dt} z^N(t) dt. \end{aligned}$$

Similarly,

$$\begin{aligned} \sum_{i=0}^N w_i \frac{z_i}{g(c_i)} \sum_{j=0}^N D_{ij} p(c_j) &= \int_{-1}^1 g(t) P_N\left(\frac{z^N}{g}\right)(t) \frac{d}{dt} P_N(p)(t) dt \\ &= \int_{-1}^1 z^N(t) \frac{d}{dt} p(t) dt + \int_{-1}^1 g(t) \left(P_N\left(\frac{z^N}{g}\right)(t) - \frac{z^N(t)}{g(t)} \right) \frac{d}{dt} P_N(p)(t) dt \\ &\quad + \int_{-1}^1 g(t) \frac{z^N(t)}{g(t)} \frac{d}{dt} (P_N(p)(t) - p(t)) dt. \end{aligned}$$

These identities imply (3.22).

Equation (3.23) follows from the choice $z_j = \delta_{jk}$. \square

Remark 3.4 If $g = 1$, then

$$P_N\left(\frac{z^N}{g}\right) - \frac{z^N}{g} = P_N(z^N) - z^N = 0$$

for all z_0, \dots, z_N .

The properties of the Lagrange polynomials imply

$$P_N\left(\frac{\psi_k}{g}\right)(t) = \sum_{i=0}^N \frac{\psi_k(c_i)}{g(c_i)} \psi_i(t) = \frac{\psi_k(t)}{g(c_k)}.$$

Remark 3.5 The integrals in (3.22) and (3.24) are well defined for $g(t) = 1$ and $g(t) = 1/\sqrt{1-t^2}$. This is obvious for $g(t) = 1$. In the other case it can be seen from the following argument. Let $g(t) = 1/\sqrt{1-t^2}$ and let $h : [-1, 1] \rightarrow \mathbb{R}$ be a continuously differentiable function with $h(\pm 1) = 0$. By the l'Hospital rule,

$$\begin{aligned} \lim_{t \rightarrow \pm 1} h(t)g(t) &= \lim_{t \rightarrow \pm 1} \frac{h(t)}{1/g(t)} \\ &= \lim_{t \rightarrow \pm 1} \frac{h'(t)}{-g'(t)/g^2(t)} \\ &= \lim_{t \rightarrow \pm 1} h'(t) \frac{\sqrt{1-t^2}}{t} = 0. \end{aligned}$$

Hence the integrands in (3.22) and (3.24) are bounded on $[-1, 1]$ and continuous on $(-1, 1)$.

The next lemma shows that the adjoint variable p divided by the weighting function g satisfies the weighted-discrete adjoint equations (3.19a) with an error that is dependent on the true adjoint p and on N .

Lemma 3.2 If p satisfies the adjoint equation (2.6b), and q_0 and q_f satisfy the transversality conditions (2.6c), then

$$\begin{aligned}
f_y(y(c_0), u(c_0))^T w_0 \frac{p(c_0)}{g(c_0)} - \sum_{k=0}^N D_{k0} w_k \frac{p(c_k)}{g(c_k)} &= -q_0 + r_0^a(N, p, g), \\
f_y(y(c_j), u(c_j))^T w_j \frac{p(c_j)}{g(c_j)} - \sum_{k=0}^N D_{kj} w_k \frac{p(c_k)}{g(c_k)} &= r_j^a(N, p, g), \\
&\quad j = 1, \dots, N-1, \\
f_y(y(c_N), u(c_N))^T w_N \frac{p(c_N)}{g(c_N)} - \sum_{k=0}^N D_{kN} w_k \frac{p(c_k)}{g(c_k)} &= -b_y(y(c_N))^T q_f - m_y(y(c_N)) \\
&\quad + r_N^a(N, p, g),
\end{aligned} \tag{3.25}$$

where

$$r_j^a(N, p, g) = \frac{w_j}{g(c_j)} \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right] + \epsilon_j(N, p, g), \tag{3.26}$$

with $\epsilon_j(N, p, g)$ defined in Lemma 3.1.

Proof Use equation (3.23) and the fact that p satisfies the adjoint equations (2.6b) and transversality conditions (2.6c) to deduce

$$\begin{aligned}
&f_y(y(c_j), u(c_j))^T w_j \frac{p(c_j)}{g(c_j)} - \sum_{k=0}^N D_{kj} w_k \frac{p(c_k)}{g(c_k)} \\
&= \frac{w_j}{g(c_j)} \left[f_y(y(c_j), u(c_j))^T p(c_j) + \sum_{k=0}^N D_{jk} p(c_k) \right] - p(c_N) \delta_{jN} + p(c_0) \delta_{j0} + \epsilon_j(N, p, g) \\
&= \frac{w_j}{g(c_j)} \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right] + \frac{w_j}{g(c_j)} \left[\frac{d}{dt} p(c_j) + f_y(c_j, y(c_j), u(c_j))^T p(c_j) \right] \\
&\quad - p(c_N) \delta_{jN} + p(c_0) \delta_{j0} + \epsilon_j(N, p, g) \\
&= \frac{w_j}{g(c_j)} \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right] - [m_y(y(1))^T + b_y(y(1))^T q_f] \delta_{jN} + \epsilon_j(N, p, g), \\
&\quad + q_0 \delta_{j0}
\end{aligned}$$

for $j = 0, \dots, N$. □

The residual terms (3.26) can be estimated using results from [10, 11]. To state these results we need the norms

$$\|f\|_g^2 = \int_{-1}^1 g(t) f^2(t) dt, \quad (3.27)$$

and

$$\|f\|_{s,g}^2 = \sum_{k=0}^s \left\| \frac{d^k}{dt^k} f \right\|_g^2. \quad (3.28)$$

Theorem 3.6 i. Let c_0, \dots, c_N and w_0, \dots, w_N be the Chebyshev-Gauss-Lobatto collocation points and corresponding quadrature weights defined in Example 3.2 and $g(t) = 1/\sqrt{1-t^2}$. If $f : [-1, 1] \rightarrow \mathbb{R}$ is s -times continuously differentiable, then there exists a constant C independent of f and N such that

$$\|P_N(f) - f\|_{0,g} \leq CN^{-s} \|f\|_{s,g}, \quad (3.29)$$

$$\|P_N(f) - f\|_{1,g} \leq CN^{2-s} \|f\|_{s,g}, \quad (3.30)$$

and

$$\left(\sum_{j=0}^N w_j \left(\frac{d}{dt} P_N(f)(c_j) - \frac{d}{dt} f(c_j) \right)^2 \right)^{1/2} \leq CN^{2-s} \|f\|_{s,g}. \quad (3.31)$$

ii. Let c_0, \dots, c_N and w_0, \dots, w_N be the Legendre-Gauss-Lobatto collocation points and corresponding quadrature weights defined in Example 3.3 and $g(t) = 1$. If $f : [-1, 1] \rightarrow \mathbb{R}$ is s -times continuously differentiable, then there exists a constant C independent of f and N such that

$$\|P_N(f) - f\|_{0,1} \leq CN^{1/2-s} \|f\|_{s,1}, \quad (3.32)$$

$$\|P_N(f) - f\|_{1,1} \leq CN^{5/2-s} \|f\|_{s,1}, \quad (3.33)$$

and

$$\left(\sum_{j=0}^N w_j \left(\frac{d}{dt} P_N(f)(c_j) - \frac{d}{dt} f(c_j) \right)^2 \right)^{1/2} \leq CN^{5/2-s} \|f\|_{s,1}. \quad (3.34)$$

Proof Estimates (3.29)–(3.31) can be found in [10, p.298]. Estimates (3.32)–(3.34) can be found in [10, pp.293/294]. \square

Corollary 3.1 i. Let c_0, \dots, c_N be the Chebyshev-Gauss-Lobatto collocation points defined in Example 3.2 and $g(t) = 1/\sqrt{1-t^2}$. If $f : [-1, 1] \rightarrow \mathbb{R}$ is s -times continuously differentiable, $s > 2$, and $\sigma > 0$ then there exists a constant C independent of f and N such that

$$\begin{aligned} \left(\sum_{j=0}^N (r_j^a(N, f, g))^2 \right)^{1/2} &\leq CN^{2-s} \left(\left\| \frac{f}{g} \right\|_{s,g} + \|f\|_{s,g} \right) \\ &\quad + CN^{-\sigma} \|f\|_{s,g} \max_{0 \leq j \leq N} \left\| \frac{\psi_j}{g} \right\|_{\sigma,g}. \end{aligned} \quad (3.35)$$

ii. Let c_0, \dots, c_N be the Legendre-Gauss-Lobatto collocation points defined in Example 3.3 and $g(t) = 1$. If $f : [-1, 1] \rightarrow \mathbb{R}$ is s -times continuously differentiable, $s > 5/2$, then there exists a constant C independent of f and N such that

$$\left(\sum_{j=0}^N (r_j^a(N, f, 1))^2 \right)^{1/2} \leq CN^{5/2-s} \|f\|_{s,1}. \quad (3.36)$$

Proof In this proof $C > 0$ denotes a generic constraint independent of N and f . The Cauchy-Schwarz inequality yields

$$\begin{aligned} \epsilon_j(N, f, g) &\leq \left\| P_N \left(\frac{f}{g} \right) - \frac{f}{g} \right\|_{0,g} \|\psi_j\|_{1,g} + \left\| P_N \left(\frac{\psi_j}{g} \right) - \frac{\psi_j}{g} \right\|_{0,g} \|P_N(f)\|_{1,g} \\ &\quad + \left\| \frac{\psi_j}{g} \right\|_{0,g} \|P_N(f) - f\|_{1,g}. \end{aligned}$$

Hence,

$$\begin{aligned} \sum_{j=0}^N \epsilon_j^2(N, f, g) &\leq 2 \left\| P_N \left(\frac{f}{g} \right) - \frac{f}{g} \right\|_{0,g}^2 \sum_{j=0}^N \|\psi_j\|_{1,g}^2 \\ &\quad + 4 \sum_{j=0}^N \left\| P_N \left(\frac{\psi_j}{g} \right) - \frac{\psi_j}{g} \right\|_{0,g}^2 \|P_N(f)\|_{1,g}^2 \\ &\quad + 4 \|P_N(f) - f\|_{1,g}^2 \sum_{j=0}^N \left\| \frac{\psi_j}{g} \right\|_{0,g}^2. \end{aligned}$$

There exists a $C > 0$ independent of N such that the inverse estimate

$$\left\| \frac{d}{dt} \psi_j \right\|_{0,g} \leq C N^2 \|\psi_j\|_{0,g}$$

holds (see equations (9.4.4) and (9.5.4) in [10]). Furthermore, there exists a constant $C > 0$ such that

$$\|\psi_j\|_{0,g} \leq C \left(\sum_{i=0}^N w_i \psi_j^2(c_i) \right)^{1/2} = C \sqrt{w_j}$$

(see equation (9.3.2) in [10]). Finally,

$$\left\| \frac{\psi_j}{g} \right\|_{0,g} \leq \|1/g\|_\infty \|\psi_j\|_{0,g} \leq \|\psi_j\|_{0,g} \leq C \sqrt{w_j}.$$

Using $w_j \in (0, 1)$ and $\sum_{i=0}^N w_i = 1$, the previous inequalities imply the existence of a constant C independent of N such that

$$\sum_{j=0}^N \|\psi_j\|_{1,g}^2 \leq C N^4, \quad \sum_{j=0}^N \left\| \frac{\psi_j}{g} \right\|_{0,g}^2 \leq C.$$

Consequently, there exists $C > 0$ with

$$\begin{aligned} \sum_{j=0}^N \epsilon_j^2(N, f, g) &\leq C N^4 \left\| P_N \left(\frac{f}{g} \right) - \frac{f}{g} \right\|_{0,g}^2 \\ &\quad + C \|P_N(f)\|_{1,g}^2 \sum_{j=0}^N \left\| P_N \left(\frac{\psi_j}{g} \right) - \frac{\psi_j}{g} \right\|_{0,g}^2 \\ &\quad + C \|P_N(f) - f\|_{1,g}^2. \end{aligned} \tag{3.37}$$

If $g = 1$, then $P_N(\psi_j/g) - \psi_j/g = P_N(\psi_j) - \psi_j = 0$, $j = 0, \dots, N$.

The inequality $\|P_N(f)\|_{1,g} \leq \|f\|_{1,g} + \|P_N(f) - f\|_{1,g}$ and (3.30), (3.33) imply

$$\|P_N(f)\|_{1,g} \leq C \|f\|_{s,g} \quad \forall N.$$

Using $w_j/g^2(c_j) \leq 1$, we find that

$$\begin{aligned} &\sum_{j=0}^N \frac{w_j^2}{g^2(c_j)} \left[\frac{d}{dt} P_N(f)(c_j) - \frac{d}{dt} f(c_j) \right]^2 \\ &\leq \sum_{j=0}^N \frac{w_j}{g^2(c_j)} w_j \left[\frac{d}{dt} P_N(f)(c_j) - \frac{d}{dt} f(c_j) \right]^2, \\ &\leq \sum_{j=0}^N w_j \left[\frac{d}{dt} P_N(f)(c_j) - \frac{d}{dt} f(c_j) \right]^2. \end{aligned} \tag{3.38}$$

The desired estimates now follow from (3.37), (3.38) and Theorem 3.6. \square

The following consistency result is an immediate consequence of Lemma 3.2.

Lemma 3.3 (Consistency) Let p satisfy the adjoint equation (2.6b), let q_0 and q_f satisfy the transversality conditions (2.6c), and let λ_j , $j = 0, \dots, N$, μ_0, μ_N satisfy (3.19a). If $f_y(y(c_j), u(c_j)) = f_y(y_j, u_j)$, $j = 0, \dots, N$, $b_y(y(c_N)) = b_y(y_N)$ and $m_y(y(c_N)) = m_y(y_N)$, then

$$\begin{aligned} & f_y(y(c_0), u(c_0))^T w_0 \left(\frac{p(c_0)}{g(c_0)} - \lambda_0 \right) - \sum_{k=0}^N D_{k0} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_k \right) \\ &= -(q_0 - \mu_0) + r_0^a(N, p, g), \\ & f_y(y(c_j), u(c_j))^T w_j \left(\frac{p(c_j)}{g(c_j)} - \lambda_j \right) - \sum_{k=0}^N D_{kj} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_k \right) \\ &= r_j^a(N, p, g) \quad j = 1, \dots, N-1, \\ & f_y(y(c_N), u(c_N))^T w_N \left(\frac{p(c_N)}{g(c_N)} - \lambda_N \right) - \sum_{k=0}^N D_{kN} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_k \right) \\ &= -b_y(y(c_N))^T (q_f - \mu_N) - m_y(y(c_N)) + r_N^a(N, p, g) \end{aligned}$$

where $r_j^a(N, p, g)$, $j = 0, \dots, N$, is defined in (3.26).

Proof This result follows immediately by subtracting the weighted discrete adjoint equations (3.19a) from (3.25). \square

Note that since the discretized optimal control problem (3.5) has $n_y(N+2) + n_b$ constraints, but only $n_y(N+1)$ state variables, there are only $n_y(N+1)$ discrete adjoint equations for the $n_y(N+2) + n_b$ Lagrange multipliers $\tilde{\lambda}_0, \dots, \tilde{\lambda}_N, \mu_0, \mu_N$. Hence, the Lagrange multipliers cannot be computed from (3.19a) alone. Therefore, it is not possible to use Lemma 3.3 and a stability result to obtain an estimate for the error between $p(c_j)/g(c_j)$ and $\tilde{\lambda}_j/w_j$. Such an estimate will be obtained in Section 3.4, where the entire optimality system is considered.

The following example illustrates the adjoint estimate (3.18) applied to the orbit transfer problem, Example 2.3.

Example 3.7 Consider the Example 2.3. We apply a Legendre and a Chebyshev pseudospectral discretization, with $N = 100$, to this problem. Figure 3.2 shows the Lagrange multipliers $\tilde{\lambda}_j$ as well as estimated adjoint variables $p(c_j) \approx g(c_j)\tilde{\lambda}_j/w_j$ for each discretization. Since the weighting function $g(t) = 1/\sqrt{1-t^2}$ for the Chebyshev pseudospectral methods is singular at ± 1 , the estimated adjoint $g(c_j)\tilde{\lambda}_j/w_j$ becomes less accurate as $t \rightarrow \pm 1$.

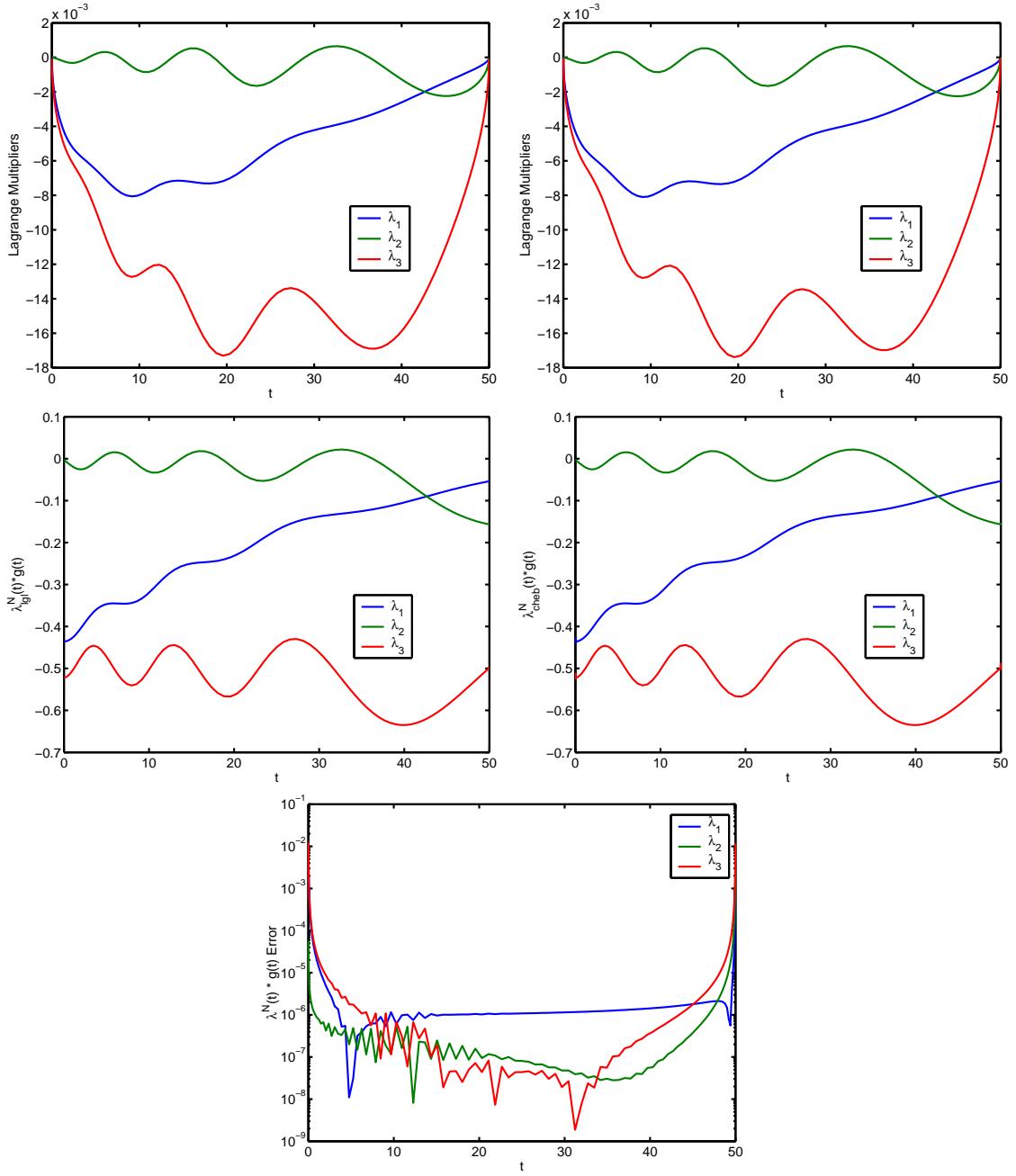


Figure 3.2: Lagrange Multipliers $\tilde{\lambda}^N$ and Estimated Adjoints $g\lambda^N$. Top Left: Legendre Pseudospectral Lagrange Multipliers $\tilde{\lambda}^N$. Top Right: Chebyshev Pseudospectral Lagrange Multipliers $\tilde{\lambda}^N$. Mid Left: Legendre Pseudospectral Adjoint Estimates. Mid Right: Chebyshev Pseudospectral Adjoint Estimates. Bottom Middle: Error Between Legendre Adjoint Estimates and Chebyshev Adjoint Estimates

3.3.2 Weighting Matrix Minimization Approach

In the previous section, we have obtained the consistency result in Lemma 3.3 by rewriting the adjoint equations (2.6b) evaluated at c_j , $j = 0, \dots, N$, in the form of the weighted discrete adjoint equations (3.19a) using the discrete integration by parts formula (3.22).

Alternatively, one may consider an approach which is motivated by the adjoint estimation procedure in [17]. In this approach, we identify

$$\frac{\tilde{\lambda}_j}{\tilde{w}_j} \approx p(c_j), \quad j = 0, \dots, N, \quad (3.39)$$

where \tilde{w}_j , $j = 0, \dots, N$ are suitably chosen weights to be determined below.

Let $\tilde{w}_j \neq 0$, $j = 0, \dots, N$, and consider the identity

$$\begin{aligned} & f_y(y_j, u_j)^T \tilde{\lambda}_j - \sum_{k=0}^N D_{kj} \tilde{\lambda}_k \\ &= f_y(y_j, u_j)^T \tilde{\lambda}_j - \sum_{k=0}^N \tilde{w}_k D_{kj} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \\ &= f_y(y_j, u_j)^T \tilde{\lambda}_j - \sum_{k=0}^N (\tilde{w}_k D_{kj} + \tilde{w}_j D_{jk}) \frac{\tilde{\lambda}_k}{\tilde{w}_k} + \sum_{k=0}^N \tilde{w}_j D_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \\ &= \tilde{w}_j \left[f_y(y_j, u_j)^T \frac{\tilde{\lambda}_j}{\tilde{w}_j} + \sum_{k=0}^N D_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right] - \sum_{k=0}^N (\tilde{w}_k D_{kj} + \tilde{w}_j D_{jk}) \frac{\tilde{\lambda}_k}{\tilde{w}_k}. \end{aligned} \quad (3.40)$$

If

$$\tilde{w}_k D_{kj} + \tilde{w}_j D_{jk} = \begin{cases} -1 & j = k = 0, \\ 1 & j = k = N, \\ 0 & \text{otherwise}, \end{cases} \quad (3.41)$$

then we will show below that $\tilde{\lambda}_j/\tilde{w}_j$ and $p(c_j)$ are related. However, the identities (3.41) cannot always be satisfied. Therefore, let $E \in \mathbb{R}^{(N+1) \times (N+1)}$ be the matrix with entries

$$E_{jk} = \begin{cases} -1 & j = k = 0, \\ 1 & j = k = N, \\ 0 & \text{otherwise} \end{cases} \quad (3.42)$$

and choose the \tilde{w}_j 's such that

$$\sum_{j,k=0}^N (\tilde{w}_k D_{kj} + \tilde{w}_j D_{jk} - E_{jk})^2 = \|\tilde{W}D + D^T\tilde{W} - E\|_F^2,$$

is minimized, where $\tilde{W} = \text{diag}(\tilde{w}_0, \dots, \tilde{w}_N)$ and $\|\cdot\|_F$ is the Frobenius norm. The problem

$$\min_{\tilde{w}_0, \dots, \tilde{w}_N} \|\tilde{W}D + D^T\tilde{W} - E\|_F^2, \quad (3.43)$$

is a linear least squares problem.

Remark 3.8 For any choice of collocation points for which the corresponding differentiation matrix D has a nonzero diagonal entry D_{jj} , $j \in \{1, \dots, N-1\}$, there is no $\tilde{w}_j \neq 0$ such that

$$\tilde{w}_j D_{jj} = -\tilde{w}_j D_{jj}. \quad (3.44)$$

Consequently, in this case there are no $\tilde{w}_j \neq 0$, $j = 1, \dots, N-1$, for which $\tilde{W}D + D^T\tilde{W} - E = 0$.

The differentiation matrix (3.7) for the Chebyshev pseudospectral method, satisfies $D_{jj} \neq 0$, $j = 0, \dots, N$.

The differentiation matrix for the Legendre pseudospectral method, which uses Legendre-Gauss-Lobatto points, satisfies $D_{11} = \dots = D_{N-1,N-1} = 0$ (see Example 3.3). In this case the solution of the linear least squares problem (3.43) is known and satisfies (3.41).

Lemma 3.4 Let c_j , $j = 0, \dots, N$, be the Legendre-Gauss-Lobatto points and let D be the corresponding differentiation matrix (3.10). If

$$\tilde{w}_j = w_j = \frac{2}{N(N+1)} \frac{1}{L_N^2(c_j)}, \quad (3.45)$$

then

$$\begin{aligned}
 \tilde{w}_j D_{jk} &= -\tilde{w}_k D_{kj}, \quad j \neq k, \\
 \tilde{w}_j D_{jj} &= -\tilde{w}_j D_{jj}, \quad j = 1, \dots, N-1, \\
 2D_{00} &= -1/\tilde{w}_0, \\
 2D_{NN} &= 1/\tilde{w}_N.
 \end{aligned} \tag{3.46}$$

Proof This result can easily be verified, keeping in mind that $L_N(-1) = (-1)^N$, $L_N(1) = 1$. \square

For the Chebyshev collocation the following lemma provides a suboptimal solution of the linear least squares problem (3.43).

Lemma 3.5 Let

$$c_j = -\cos\left(\frac{j\pi}{N}\right), \quad j = 0, \dots, N,$$

(Chebyshev-Gauss-Lobatto collocation points), let D be the corresponding differentiation matrix (3.7). If

$$\tilde{w}_j = w_j = \begin{cases} \frac{\pi}{2N} & j = 0, N, \\ \frac{\pi}{N} & j = 1, \dots, N-1, \end{cases} \tag{3.47}$$

then

$$\begin{aligned}
 \tilde{w}_j D_{jk} &= -\tilde{w}_k D_{kj}, \quad j \neq k, \\
 2D_{00} &= -1/\tilde{w}_0, \\
 2D_{NN} &= 1/\tilde{w}_N.
 \end{aligned} \tag{3.48}$$

The least squares norms $\|\widetilde{W}D + D^T\widetilde{W} - E\|_F^2$, using the weights defined by (3.47), for different numbers of collocation points are shown in Figure 3.3.

For the Chebyshev collocation points the linear least squares problem (3.43) is solved numerically. The least squares norms $\|\widetilde{W}D + D^T\widetilde{W} - E\|_F^2$, using optimal weights, for different numbers of collocation points are shown in Figure 3.4. Figure 3.5

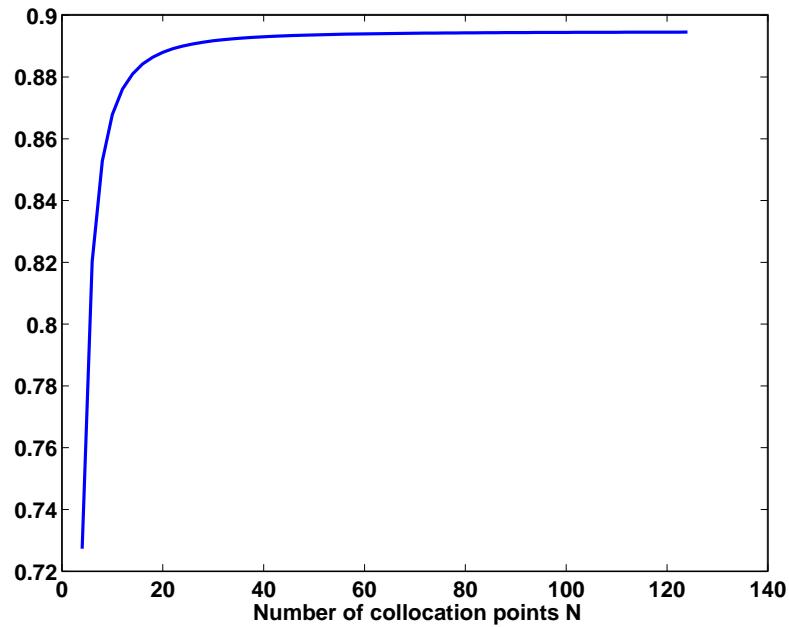


Figure 3.3: The least squares norms $\|\widetilde{W}D + D^T\widetilde{W} - E\|_F^2$ for \tilde{w} given by (3.47) for different numbers of collocation points

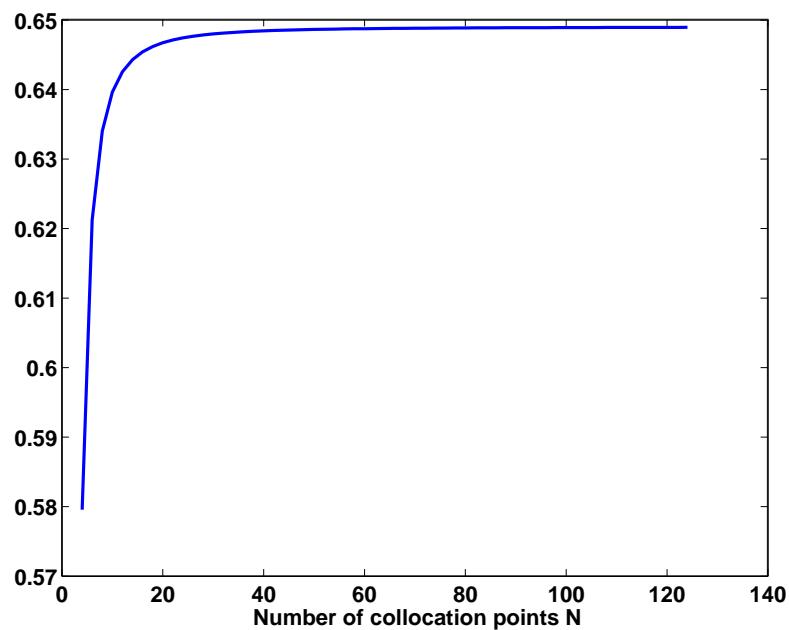


Figure 3.4: The least squares norms $\|\widetilde{W}D + D^T\widetilde{W} - E\|_F^2$ for different numbers of collocation points

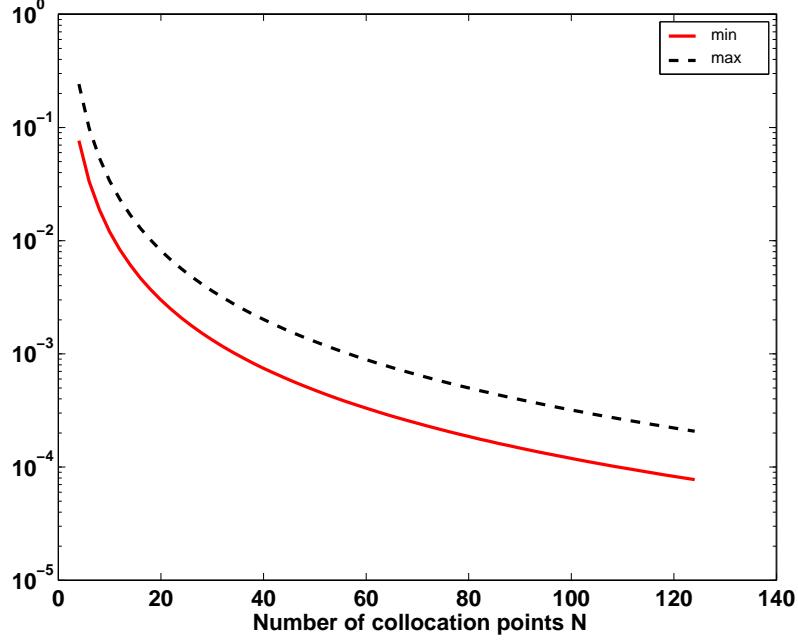


Figure 3.5: $\min \tilde{w}_i$ and $\max \tilde{w}_i$ for different numbers of collocation points

shows that the optimal weights \tilde{w}_j , $j = 0, \dots, N$, are positive. The element wise error $|\widetilde{W}D + D^T\widetilde{W} - E|$ for $N = 64$ is displayed in Figure 3.6.

If we define

$$\epsilon_{jk} = \tilde{w}_k D_{kj} + \tilde{w}_j D_{jk} - E_{jk}, \quad (3.49)$$

$j, k = 0, \dots, N$, and use the identities (3.40) in (3.13a), we obtain

$$\begin{aligned} \tilde{w}_0 & \left[f_y(y_0, u_0)^T \frac{\tilde{\lambda}_0}{\tilde{w}_0} + \sum_{k=0}^N D_{0k} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right] \\ & + \frac{\tilde{\lambda}_0}{\tilde{w}_0} + \mu_0 = \sum_{k=0}^N \epsilon_{0k} \frac{\tilde{\lambda}_k}{\tilde{w}_k}, \\ \tilde{w}_j & \left[f_y(y_j, u_j)^T \frac{\tilde{\lambda}_j}{\tilde{w}_j} + \sum_{k=0}^N D_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right] = \sum_{k=0}^N \epsilon_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k}, \quad j = 1, \dots, N-1, \quad (3.50) \\ \tilde{w}_N & \left[f_y(y_N, u_N)^T \frac{\tilde{\lambda}_N}{\tilde{w}_N} + \sum_{k=0}^N D_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right] \\ & - \frac{\tilde{\lambda}_N}{\tilde{w}_N} + m_y(y_N) + b_y(y_N)^T \mu_N = \sum_{k=0}^N \epsilon_{Nk} \frac{\tilde{\lambda}_k}{\tilde{w}_k}. \end{aligned}$$

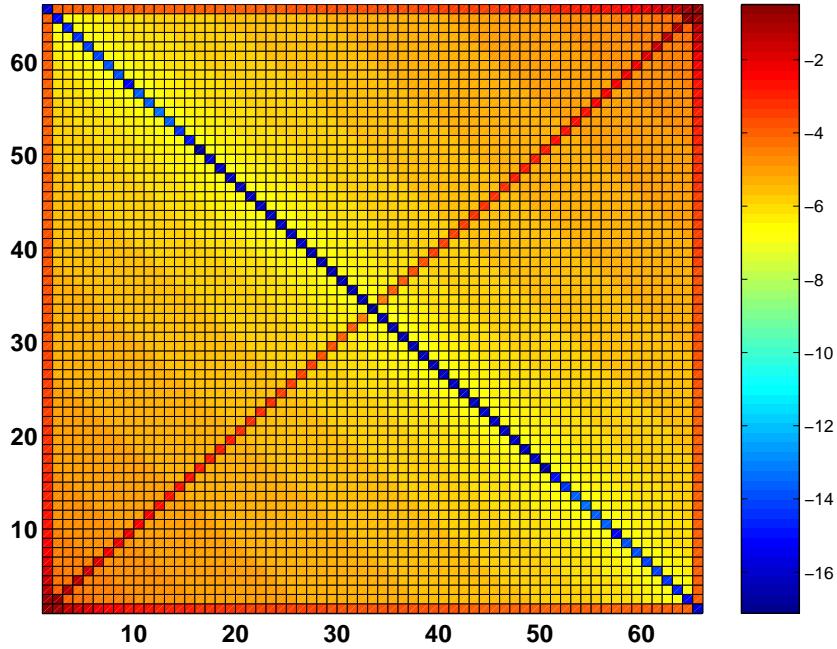


Figure 3.6: Element Wise Error $\log_{10} |(\widetilde{W}D + D^T \widetilde{W} - E)_{ij}|$ for $N = 64$.

We use the adjoint equations (2.6b) evaluated at c_j , $j = 0, \dots, N$, and the transversality conditions (2.6c), to obtain

$$\begin{aligned}
& \widetilde{w}_0 \left[f_y(y(c_0), u(c_0))^T p(c_0) + \sum_{k=0}^N D_{0k} p(c_k) \right] \\
& \quad + p(c_0) + q_0 = \widetilde{w}_0 \left[\frac{d}{dt} P_N(p)(c_0) - \frac{d}{dt} p(c_0) \right], \\
& \widetilde{w}_j \left[f_y(y(c_j), u(c_j))^T p(c_j) + \sum_{k=0}^N D_{jk} p(c_k) \right] = \widetilde{w}_j \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right], \\
& \quad j = 1, \dots, N-1, \\
& \widetilde{w}_N \left[f_y(y(c_N), u(c_N))^T p(c_N) + \sum_{k=0}^N D_{jk} p(c_k) \right] \\
& \quad - p(c_N) + m_y(y(c_N)) + b_y(y(c_N))^T q_f = \widetilde{w}_N \left[\frac{d}{dt} P_N(p)(c_N) - \frac{d}{dt} p(c_N) \right].
\end{aligned} \tag{3.51}$$

Recall the definition (3.20) of P_N .

Subtracting (3.50) from (3.51) leads to the following result.

Lemma 3.6 (Consistency) Let p satisfy the adjoint equation (2.6b), let q_0 and q_f satisfy the transversality conditions (2.6c), and let λ_j , $j = 0, \dots, N$, μ_0, μ_N satisfy (3.19a). If $f_y(y(c_j), u(c_j)) = f_y(y_j, u_j)$, $j =$

$0, \dots, N$, $b_y(y(c_N)) = b_y(y_N)$ and $m_y(y(c_N)) = m_y(y_N)$, then

$$\begin{aligned}
& \tilde{w}_0 \left[f_y(y(c_0), u(c_0))^T \left(p(c_0) - \frac{\tilde{\lambda}_0}{\tilde{w}_0} \right) + \sum_{k=0}^N D_{0k} \left(p(c_k) - \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right) \right] \\
& + \left(p(c_0) - \frac{\tilde{\lambda}_0}{\tilde{w}_0} \right) + q_0 - \mu_0 \\
= & \tilde{w}_0 \left[\frac{d}{dt} P_N(p)(c_0) - \frac{d}{dt} p(c_0) \right] - \sum_{k=0}^N \epsilon_{0k} \frac{\tilde{\lambda}_k}{\tilde{w}_k}, \\
& \tilde{w}_j \left[f_y(y(c_j), u(c_j))^T \left(p(c_j) - \frac{\tilde{\lambda}_j}{\tilde{w}_j} \right) + \sum_{k=0}^N D_{jk} \left(p(c_k) - \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right) \right] \\
= & \tilde{w}_j \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right] \sum_{k=0}^N \epsilon_{jk} \frac{\tilde{\lambda}_k}{\tilde{w}_k}, \quad j = 1, \dots, N-1, \\
& \tilde{w}_N \left[f_y(y(c_N), u(c_N))^T \left(p(c_N) - \frac{\tilde{\lambda}_N}{\tilde{w}_N} \right) + \sum_{k=0}^N D_{jk} \left(p(c_k) - \frac{\tilde{\lambda}_k}{\tilde{w}_k} \right) \right] \\
& - \left(p(c_N) - \frac{\tilde{\lambda}_N}{\tilde{w}_N} \right) + m_y(y(c_N)) + b_y(y(c_N))^T (q_f - \mu_N) \\
= & \tilde{w}_N \left[\frac{d}{dt} P_N(p)(c_N) - \frac{d}{dt} p(c_N) \right] - \sum_{k=0}^N \epsilon_{Nk} \frac{\tilde{\lambda}_k}{\tilde{w}_k}.
\end{aligned} \tag{3.52}$$

In the case of Legendre-Gauss-Lobatto points, c_j , $j = 0, \dots, N$, the weights

$$\tilde{w}_j = w_j = \frac{2}{N(N+1)} \frac{1}{L_N^2(c_j)},$$

are optimal and lead to

$$\epsilon_{jk} = 0, \quad j, k = 0, \dots, N.$$

In this case the adjoint estimates (3.18) and (3.39) are identical. Furthermore, in this case the consistency results in Lemma 3.3 and in Lemma 3.6 are identical. However, for the Chebyshev-Gauss-Lobatto points, Remark 3.8 shows that

$$|\epsilon_{jj}| > 0, \quad j = 0, \dots, N,$$

and the numerical results displayed in Figure 3.4 indicate that

$$\sum_{j=0}^N \sum_{k=0}^N \epsilon_{jk}^2 \rightarrow \epsilon_* \approx 0.65 > 0 \quad (N \rightarrow \infty).$$

Hence, in the case of Chebyshev-Gauss-Lobatto collocation, the adjoint estimate (3.39) suggested by the weighting matrix approach is not useful, unlike the adjoint estimate (3.18) derived earlier.

3.4 Discretization Error for the Optimal Control

With the adjoint estimation procedure in place, it is now possible to quantify the error between the state

$$y^N(t) = \sum_{i=0}^N y_i \psi_i(t),$$

control

$$u^N(t) = \sum_{i=0}^N u_i \psi_i(t),$$

and adjoint

$$\lambda^N(t) = \sum_{i=0}^N \lambda_i \psi_i(t),$$

computed as the optimal solution of the discretized optimal control problem (3.5) and the solution y , u , and p of the infinite dimensional optimal control problem (2.5).

Recall that y_j , u_j , λ_j , $j = 0, \dots, N$ and μ_0, μ_N satisfy the weighted discrete adjoint equations

$$\begin{aligned} f_y(y_0, u_0)^T w_0 \lambda_0 - \sum_{k=0}^N D_{k,0} w_k \lambda_k &= -\mu_0, \\ f_y(y_j, u_j)^T w_j \lambda_j - \sum_{k=0}^N D_{k,j} w_k \lambda_k &= 0, \quad j = 1, \dots, N-1, \\ f_y(y_N, u_N)^T w_N \lambda_N - \sum_{k=0}^N D_{k,N} w_k \lambda_k &= -b_y(y_N)^T \mu_N - m_y(y_N), \end{aligned} \tag{3.53a}$$

the weighted-discrete gradient equations

$$f_u(y_j, u_j)^T w_j \lambda_j = 0, \quad j = 0, \dots, N, \tag{3.53b}$$

and weighted-discretized state equations

$$\begin{aligned} w_j \left(f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right) &= 0, \quad j = 0, \dots, N, \\ y_0 - \bar{y}_0 &= 0, \\ b(y_N) &= 0. \end{aligned} \tag{3.53c}$$

If f is affine linear,

$$f(y(t), u(t)) = F_y(t)y(t) + F_u(t)u(t) + f_a(t),$$

if m is quadratic,

$$m(y(t_f)) = \frac{1}{2}y(t_f)^T M y(t_f) + m_l^T y(t_f) + m_a,$$

and if b is affine linear

$$b(y(t_f)) = B y(t_f) + b_a,$$

then the optimality conditions (3.53) can be written as

$$\mathbf{K}_N \mathbf{x}_N = \mathbf{b}_N, \tag{3.54}$$

where

$$\mathbf{x}_N = (y_0^T, \dots, y_N^T, u_0^T, \dots, u_N^T, \lambda_0^T, \dots, \lambda_N^T, \mu_0^T, \mu_N^T)^T. \tag{3.55}$$

Lemma 3.7 If y, u satisfy the state equations (2.5b)–(2.5d), then

$$\begin{aligned} w_j \left[f(y(c_j), u(c_j)) - \sum_{k=0}^N D_{j,k} y(c_k) \right] &= r_j^s(N, y, 1), \quad j = 0, \dots, N, \\ y(c_0) - \bar{y}_0 &= 0, \\ b(y(c_N)) &= 0, \end{aligned} \tag{3.56}$$

where

$$r_j^s(N, p, g) = \frac{w_j}{g(c_j)} \left[\frac{d}{dt} P_N(p)(c_j) - \frac{d}{dt} p(c_j) \right]. \tag{3.57}$$

Proof This result follows from evaluating (2.5b) at the collocation points and using the definition (3.20) of the interpolating polynomial. \square

The following error results are shown for linear-quadratic optimal control problems. More analysis is needed to extend these results to nonlinear OCPs, however that exceeds the scope of this thesis.

The following lemma provides a consistency result which will be used to derive an error estimate for the optimal control for linear-quadratic OCPs.

Lemma 3.8 (*Consistency for Linear-Quadratic OCPs*) Let f be affine linear,

$$f(y(t), u(t)) = F_y(t)y(t) + F_u(t)u(t) + f_a(t),$$

let m be quadratic,

$$m(y(t_f)) = \frac{1}{2}y(t_f)^T M y(t_f) + m_l^T y(t_f) + m_a,$$

and let b be affine linear

$$b(y(t_f)) = B y(t_f) + b_a.$$

If y, u, p, q_0, q_f are the solution of (3.5) and corresponding adjoint variables and Lagrange multipliers, and if $y_0, \dots, y_N, u_0, \dots, u_N, \lambda_0, \dots, \lambda_N, \mu_0, \mu_N$ are the solution of the discretized optimal control problem (3.5) and corresponding weighted Lagrange multipliers, then

$$\begin{aligned} F_y(c_0)^T w_0 \left(\frac{p(c_0)}{g(c_0)} - \lambda_0 \right) - \sum_{k=0}^N D_{k0} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_k \right) \\ + (q_0 - \mu_0) &= r_0^a(N, p, g), \\ F_y(c_j)^T w_j \left(\frac{p(c_j)}{g(c_j)} - \lambda_j \right) - \sum_{k=0}^N D_{kj} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_j \right) &= r_j^a(N, p, g), \\ j &= 1, \dots, N-1, \\ F_y(c_N)^T w_N \left(\frac{p(c_N)}{g(c_N)} - \lambda_N \right) - \sum_{k=0}^N D_{kN} w_k \left(\frac{p(c_k)}{g(c_k)} - \lambda_N \right) \\ + B^T (q_f - \mu_N) + M(y(c_N) - y_N) &= r_N^a(N, p, g), \end{aligned} \tag{3.58a}$$

$$F_u(c_j)^T w_j \left(\frac{p(c_k)}{g(c_k)} - \lambda_j \right) = 0, \quad j = 0, \dots, N, \quad (3.58b)$$

$$\begin{aligned} w_j \left[F_y(c_j)(y(c_j) - y_j) + F_u(c_j)(u(c_j) - u_j) \right. \\ \left. - \sum_{k=0}^N D_{jk}(y(c_k) - y_k) \right] &= r_j^s(N, y, 1), \\ j &= 0, \dots, N, \\ y(c_0) - y_0 &= 0, \\ B(y(c_N) - y_N) &= 0, \end{aligned} \quad (3.58c)$$

where $r_j^a(N, y, g)$ and $r_j^s(N, y, g)$, $j = 0, \dots, N$, are defined as in (3.26) and (3.57) respectively.

Proof The equations (3.58a) were derived in Lemma 3.3. The equations (3.58b) are obtained by evaluating (2.6d) at c_j and subtracting (3.53b). The equations (3.58c) are obtained by subtracting (3.53c) from (3.56). \square

The first part of the following theorem is an immediate consequence of Lemma 3.8. parts two and three follows from Corollary 3.1.

Theorem 3.9 (*Error for Linear-Quadratic OCPs*) i. Let the assumptions of Lemma 3.8 be valid. If \mathbf{x}_N is defined as in (3.55) and if

$$\mathbf{x} = \left(y(c_0)^T, \dots, y(c_N)^T, u(c_0)^T, \dots, u(c_N)^T, \frac{p(c_0)^T}{g(c_0)}, \dots, \frac{p(c_N)^T}{g(c_N)}, q_0^T, q_f^T \right)^T,$$

then

$$\|\mathbf{x}_N - \mathbf{x}\|_2 \leq \|\mathbf{K}_N^{-1}\|_2 \|\mathbf{r}(N, y, p, g)\|_2, \quad (3.59)$$

where \mathbf{K}_N is the system matrix in (3.53), (3.54) and

$$\mathbf{r}(N, y, p, g) = \begin{pmatrix} r_0^a(N, p, g) \\ \vdots \\ r_N^a(N, p, g) \\ 0 \\ \vdots \\ 0 \\ r_0^s(N, y, 1) \\ \vdots \\ r_N^s(N, y, 1) \\ 0 \\ 0 \end{pmatrix}. \quad (3.60)$$

ii. Let c_0, \dots, c_N be the Chebyshev-Gauss-Lobatto collocation points defined in Example 3.2 and $g(t) = 1/\sqrt{1-t^2}$. If y and p are s -times continuously differentiable, $s > 2$, and $\sigma > 0$, then there exists a constant C independent of y , p and N such that

$$\begin{aligned} & \| \mathbf{r}(N, y, p, g) \|_2 \\ & \leq CN^{2-s} (\|y\|_{s,g} + \|p\|_{s,g} + \|p/g\|_{s,g}) \\ & \quad + CN^{-\sigma} \|p\|_{s,g} \max_{0 \leq j \leq N} \|\psi_j/g\|_{\sigma,g}. \end{aligned} \quad (3.61)$$

iii. Let c_0, \dots, c_N be the Legendre-Gauss-Lobatto collocation points defined in Example 3.3 and $g(t) = 1$. If y and p are s -times continuously differentiable, $s > 5/2$, then there exists a constant C independent of y , p and N such that

$$\| \mathbf{r}(N, y, p, 1) \|_2 \leq CN^{5/2-s} (\|y\|_{s,1} + \|p\|_{s,1}). \quad (3.62)$$

To obtain an error estimate, one needs a stability result that guarantees the uniform boundedness of $\|\mathbf{K}_N^{-1}\|_2$. Such a result is not yet known.

Example 3.10 Consider Example 2.13. Applying a Legendre pseudospectral discretization to this problem yields the results shown in Figure 3.7. The numerical results indicate that the solutions of the discretized problem converge quickly to the solution of the infinite dimensional problem. The lower right plot in Figure 3.7 also shows that $\|\mathbf{K}_N^{-1}\|_2$ increased significantly as N increases.

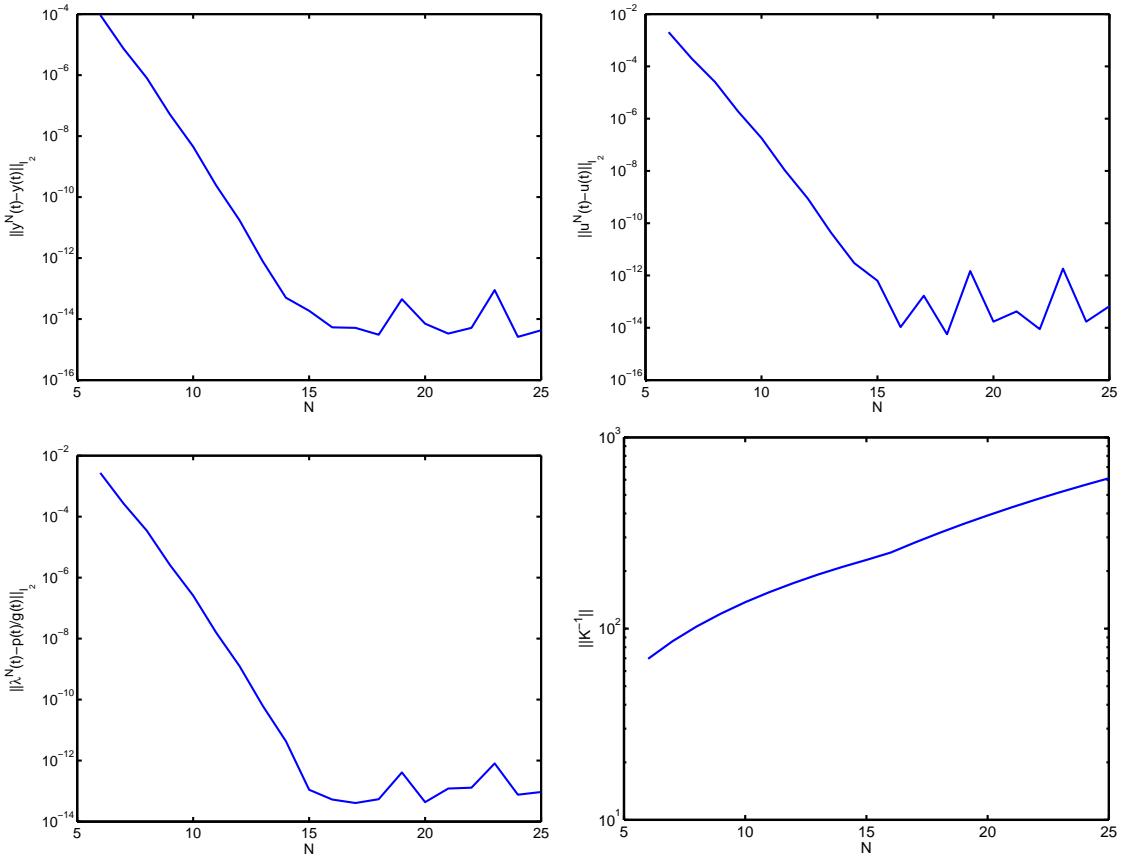


Figure 3.7: Error vs. N for Linear-Quadratic Optimal Control Problem in Mayer Form Using Legendre Pseudospectral Collocation-
Top Left: ℓ_2 State Error - Top Right: ℓ_2 Control Error
- Bottom Left: ℓ_2 Adjoint Divided by Weighting Function Error - Bottom Right: Norm of System Matrix Inverse

Example 3.11 Again consider Example 2.13. Now the Chebychev pseudospectral discretization is applied to this problem. The numerical results are shown in Figure 3.8. The error between the solutions of the discretized problem and the solution of the infinite dimensional problem decays much slower than in Example 3.10. Especially the error $\lambda^N - p/g$ for given N is much larger than in Example 3.10. We also observe that for the Chebychev pseudospectral discretization $\|\mathbf{K}_N^{-1}\|_2$ is larger and increases more rapidly as N increases than in Example 3.10.

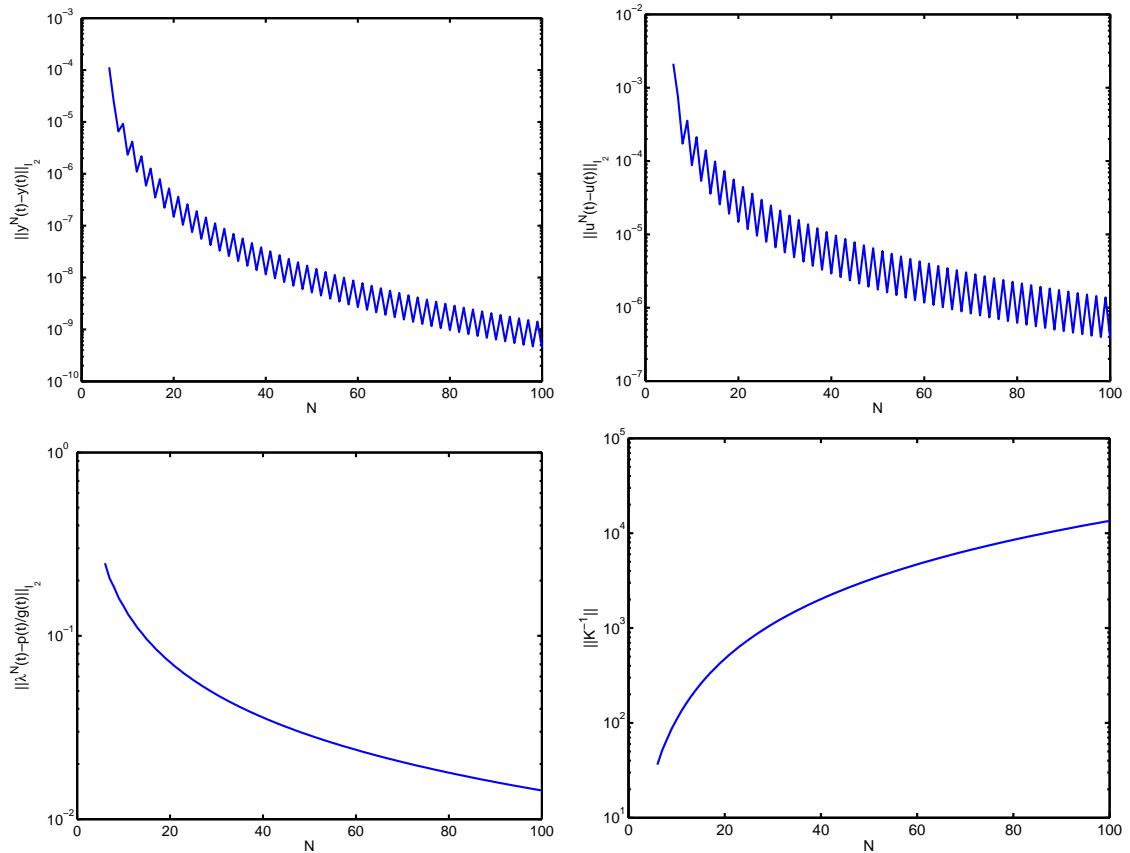


Figure 3.8: Error vs. N for Linear-Quadratic Optimal Control Problem in Mayer Form Using Chebyshev Pseudospectral Collocation- Top Left: ℓ_2 State Error - Top Right: ℓ_2 Control Error - Bottom Left: ℓ_2 Adjoint Divided by Weighting Function Error - Bottom Right: Norm of System Matrix Inverse

From the numerical results in Examples 3.10 3.11 it is questionable whether one can prove that $\|\mathbf{K}_N^{-1}\|_2$ is bounded. The numerical results, however, indicate that even if $\|\mathbf{K}_N^{-1}\|_2$ is not bounded it grows slower than $\|\mathbf{r}(N, y, p, g)\|_2$ decreases. In such a case, convergence of the solutions to the discretized problems can be guaranteed, but the rate of convergence is less than one would expect based on the consistency results alone. It is also not known whether and, if so how, the growth in $\|\mathbf{K}_N^{-1}\|_2$ is related to the increasing condition number of the constraint Jacobians reported on in Example 3.1.

3.5 Numerical Equivalence of Bolza and Mayer Forms

In this section the numerical difference between the Bolza form OCP (2.1) and the Mayer form OCP (2.9) is addressed. In [17] the argument is made that for the Legendre pseudospectral method the quadrature rule used to compute the integral in (2.1a) is equivalent to the resulting auxiliary discrete adjoint equations in the transformed problem (2.9). This would imply that for the Legendre pseudospectral method, a direct transcription of either problem leads to the same numerical solution. It will be shown that this assertion is not quite correct and that solving the discretized OCP in Mayer and Bolza forms, respectively, yield results that merely converge to the same solution as $N \rightarrow \infty$. Recall the Bolza form optimal control problem

$$\min \quad m(y(1)) + \int_{-1}^1 \ell(y(t), u(t)) dt, \quad (3.63a)$$

s.t.

$$\frac{d}{dt} y(t) = f(y(t), u(t)), \quad (3.63b)$$

$$y(-1) = \bar{y}_0, \quad (3.63c)$$

$$b(y(1)) = 0. \quad (3.63d)$$

Recall the transformed Mayer form OCP

$$\min \quad m(y(1)) + z(1), \quad (3.64a)$$

s.t.

$$\frac{d}{dt}y(t) = f(y(t), u(t)), \quad (3.64b)$$

$$\frac{d}{dt}z(t) = \ell(y(t), u(t)), \quad (3.64c)$$

$$y(-1) = \bar{y}_0, \quad (3.64d)$$

$$z(-1) = 0, \quad (3.64e)$$

$$b(y(1)) = 0. \quad (3.64f)$$

To compare the discrete solutions to (3.63a) and (3.64) it is necessary to look at the discrete optimality systems of each. The weighted discrete Lagrangian from (3.16) for (3.64) can be written as

$$\begin{aligned} L_w(\mathbf{y}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}, \boldsymbol{\gamma}, \mu_0, \mu_N, \nu_0) &= m(y_N) + b(y_N)^T \mu_N + (y_0 - \bar{y}_0)^T \mu_0 + z_0^T \nu_0 \\ &\quad + w_j \gamma_j \left[\sum_{j=0}^N \ell(y_j, u_j) - \sum_{k=0}^N D_{j,k} z_k \right] \\ &\quad + w_j \lambda_j^T \left[\sum_{j=0}^N f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right]. \end{aligned} \quad (3.65)$$

Differentiating (3.65) with respect to the y_j 's and setting it equal to zero yields

$$\begin{aligned} f_y(y_0, u_0)^T w_0 \lambda_0 + \ell_y(y_0, u_0) w_0 \gamma_0 - \sum_{k=0}^N D_{k,0} w_k \lambda_k &= -\mu_0, \\ f_y(y_j, u_j)^T w_j \lambda_j + \ell_y(y_j, u_j) w_j \gamma_j - \sum_{k=0}^N D_{k,j} w_k \lambda_k &= 0, \quad j = 1, \dots, N-1, \\ f_y(y_N, u_N)^T w_N \lambda_N + \ell_y(y_N, u_N) w_N \gamma_N - \sum_{k=0}^N D_{k,N} w_k \lambda_k &= -b_y(y_N)^T \mu_N - m_y(y_N). \end{aligned} \quad (3.66a)$$

Differentiating (3.65) with respect to the z_j 's and setting it equal to zero yields

$$\begin{aligned} -\sum_{k=0}^N D_{k,0} w_k \gamma_k &= -\nu_0, \\ -\sum_{k=0}^N D_{k,j} w_k \gamma_k &= 0, \quad j = 1, \dots, N. \end{aligned} \quad (3.66b)$$

Differentiating the Lagrangian (3.65) with respect to the u_j 's and setting the derivatives to zero gives

$$f_u(y_j, u_j)^T w_j \lambda_j + \ell_u(y_j, u_j) w_j \gamma_j = 0, \quad j = 0, \dots, N. \quad (3.66c)$$

Differentiating the Lagrangian (3.65) with respect to the λ_j 's, μ_0 and μ_N and setting the derivatives equal to zero

$$\begin{aligned} w_j \left[f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right] &= 0, \quad j = 0, \dots, N, \\ y_0 - \bar{y}_0 &= 0, \\ b(y_N) &= 0. \end{aligned} \quad (3.66d)$$

Differentiating the Lagrangian (3.65) with respect to the λ_j 's, μ_0 and μ_N and setting the derivatives equal to zero as well

$$\begin{aligned} w_j \left[\ell(y_j, u_j) - \sum_{k=0}^N D_{j,k} z_k \right] &= 0, \quad j = 0, \dots, N, \\ z_0 &= 0. \end{aligned} \quad (3.66e)$$

Alternatively, the OCP (3.63) can be solved directly in Bolza form. In this case the integral term is approximated by

$$\begin{aligned} \int_{-1}^1 \ell(y(t), u(t)) dt &= \int_{-1}^1 \frac{g(t)}{g(t)} \ell(y(t), u(t)) dt \\ &\approx \sum_{j=0}^N \frac{w_j}{g(c_j)} \ell(y_j, u_j). \end{aligned} \quad (3.67)$$

Using (3.67), the weighted discrete Lagrangian for (3.63) can be written as

$$\begin{aligned} L_w(\mathbf{y}, \mathbf{u}, \boldsymbol{\lambda}, \mu_0, \mu_N) &= m(y_N) + \sum_{j=0}^N \frac{w_j}{g(c_j)} \ell(y_j, u_j) \\ &\quad + b(y_N)^T \mu_N + (y_0 - \bar{y}_0)^T \mu_0 \\ &\quad + w_j \lambda_j^T \left[\sum_{j=0}^N f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right]. \end{aligned} \quad (3.68)$$

Differentiating (3.68) with respect to the y_j 's and setting it equal to zero yields

$$\begin{aligned} f_y(y_0, u_0)^T w_0 \lambda_0 + \ell_y(y_0, u_0) \frac{w_0}{g(c_0)} - \sum_{k=0}^N D_{k,0} w_k \lambda_k &= -\mu_0, \\ f_y(y_j, u_j)^T w_j \lambda_j + \ell_y(y_j, u_j) \frac{w_j}{g(c_j)} - \sum_{k=0}^N D_{k,j} w_k \lambda_k &= 0, \quad j = 1, \dots, N-1, \\ f_y(y_N, u_N)^T w_N \lambda_N + \ell_y(y_N, u_N) w_N \frac{w_N}{g(c_N)} - \sum_{k=0}^N D_{k,N} w_k \lambda_k &= -b_y(y_N)^T \mu_N - m_y(y_N). \end{aligned} \quad (3.69a)$$

Differentiating the Lagrangian (3.68) with respect to the u_j 's and setting the derivatives to zero gives

$$f_u(y_j, u_j)^T w_j \lambda_j + \ell_u(y_j, u_j) \frac{w_j}{g(c_j)} = 0, \quad j = 0, \dots, N. \quad (3.69b)$$

Differentiating the Lagrangian (3.68) with respect to the λ_j 's, μ_0 and μ_N and setting the derivatives to zero gives

$$\begin{aligned} w_j \left[f(y_j, u_j) - \sum_{k=0}^N D_{j,k} y_k \right] &= 0, \quad j = 0, \dots, N, \\ y_0 - \bar{y}_0 &= 0, \\ b(y_N) &= 0. \end{aligned} \quad (3.69c)$$

Lemma 3.9 Let $y^M, u^M, \lambda^M, \gamma^M, \mu_0^M, \mu_N^M$ and ν_0^M be solutions to the weighted discrete optimality system (3.65) corresponding to the transformed Mayer form OCP (3.64). Let $y^B, u^B, \lambda^B, \mu_0^B$ and μ_N^B be solutions to the weighted discrete optimality system (3.68) corresponding to the Bolza form OCP (3.63). We define

$$\mathbf{x}_N^M = \left((y_0^M)^T, \dots, (y_N^M)^T, (u_0^M)^T, \dots, (u_N^M)^T, (\lambda_0^M)^T, \dots, (\lambda_N^M)^T, (\mu_0^M)^T, (\mu_N^M)^T \right)^T,$$

and

$$\mathbf{x}_N^B = \left((y_0^B)^T, \dots, (y_N^B)^T, (u_0^B)^T, \dots, (u_N^B)^T, (\lambda_0^B)^T, \dots, (\lambda_N^B)^T, (\mu_0^B)^T, (\mu_N^B)^T \right)^T,$$

to be the numerical solutions to (3.64) and (3.63) respectively. If the weighted discrete optimality systems (3.65) and (3.68) are sufficiently stable, then the error between these solutions, $\|\mathbf{x}_N^M - \mathbf{x}_N^B\|_2$, will converge to zero as $N \rightarrow \infty$.

Proof This result is easily verified using the adjoint estimation Lemma 3.2 and the optimality conditions (2.10) in Section 2.2, $\gamma \rightarrow r(t)/g(t) = 1/g(t)$, where $r(t)$ is the true auxiliary adjoint. \square

It is evident that the systems (3.66) and (3.69) are not equivalent. This is because the estimated auxiliary adjoint,

$$\gamma^N(t) = \sum_{k=0}^N \gamma_k \psi_k(t),$$

will only converge to its true solution $1/g(t)$ as $N \rightarrow \infty$. In order for these systems to be equivalent, the auxiliary adjoint γ^N would have to be equal to its true solution, 1, for all N . This notion is reinforced by the following example to conclude this section.

Example 3.12 Consider the example problem (2.2) in Mayer from (2.13).

A Legendre pseudospectral discretization to this problem is applied. Secondly, consider the example problem (2.2) in Bolza from (2.11). Again a Legendre pseudospectral discretization to this problem is applied. Taking the ℓ_2 norm error between the state, control, and adjoint for each N yields the results shown in Figure 3.9. Notice that the behavior described in Lemma 3.9 is exhibited. The solutions are never identical, but converge to the true solution as $N \rightarrow \infty$.

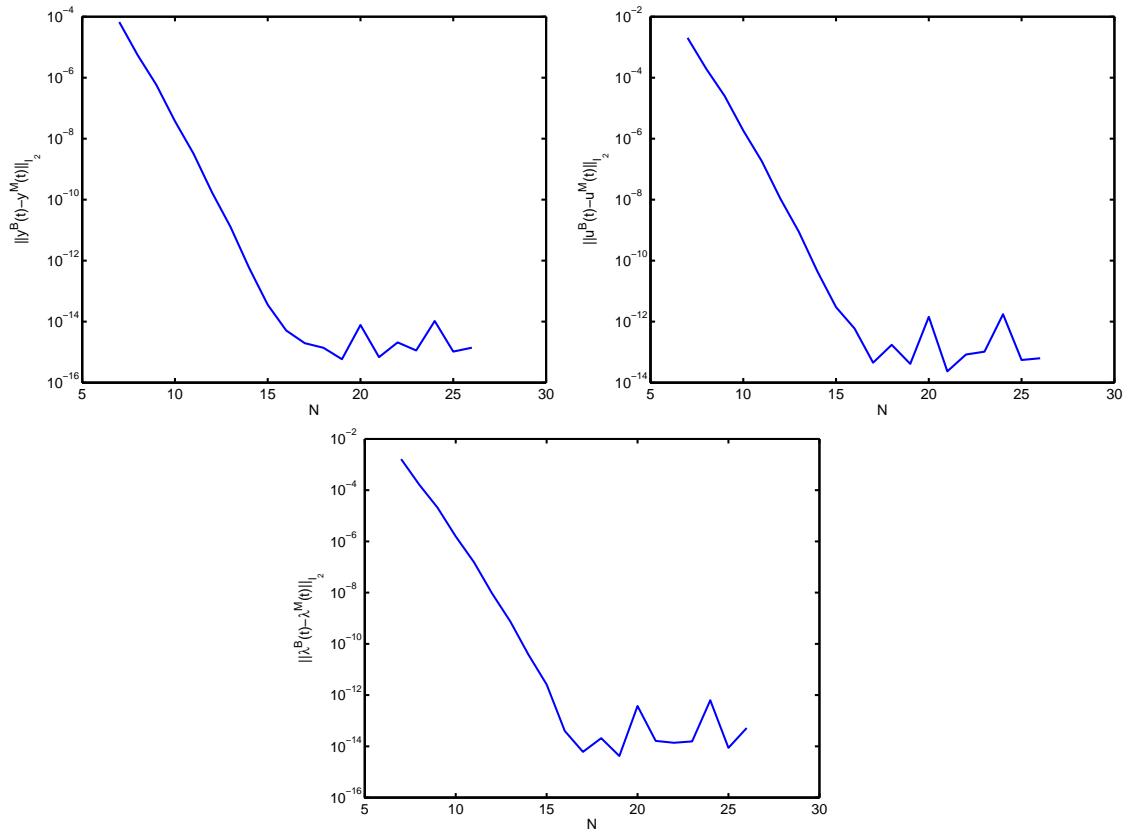


Figure 3.9: ℓ_2 Error between Mayer Form Problem and Bolza Form Problem vs. N for Linear-Quadratic Optimal Control Problem Using Legendre Pseudospectral Collocation- Top Left: State Error - Top Right: Control Error - Bottom Middle: Adjoint Divided by Weighting Function Error

3.6 Extension to Multiple Subintervals

Much of the work done in this chapter applied the pseudospectral method on one interval $[-1, 1]$. Extensions to multiple intervals are very important for many problems. Our error bound in Theorem 3.9 indicates that the discretization error between computed solution and true solution depends on the smoothness of the state and of the adjoint. The smoothness of the state depends, among other things, on the properties of the right hand side function f in the governing dynamics. For problems with piecewise continuous right hand sides (e.g., due to change of mass in launch problems, or due to piecewise constant controls), it is important to introduce multiple subintervals. Another potential benefit of using pseudospectral methods along multiple intervals is to take advantage of sparsity. Indeed the optimality system for a pseudospectral method along many subintervals will be very sparse relative to a pseudospectral method applied on one interval. The benefit is that proper exploitation of sparsity may improve solution time.

The pseudospectral method can easily be extended to multiple subintervals. To accomplish this, the collocation points

$$c_0 = -1, \quad c_1, \dots, c_{N-1} \in (-1, 1), \quad c_N = 1$$

are again used. At this point it is more useful to consider the OCP (2.5) on the interval $[t_0, t_f]$.

Remark 3.13 The time interval shift can be accomplished by the following identity. Let $t(c) \in [t_f, t_0]$ be the mapping

$$t(c) = \left((t_f - t_0)c + t_f + t_0 \right) / 2.$$

By the chain rule, we have that

$$\frac{d}{dc} y(t(c)) = \frac{d}{dt} y(t(c)) \frac{d}{dc} t(c) = \frac{t_f - t_0}{2} f(y(t(c)), u(t(c))).$$

However, for the remainder of this section, t will not be written as an explicit function of c and the more convenient notation

$$\frac{d}{dt}y(t) = \frac{t_f - t_0}{2}f(y(t), u(t)),$$

will be used.

The interval $[t_0, t_f]$ is subdivided into I subintervals $[t_i, t_{i+1}]$, $i = 0, \dots, I-1$, with

$$t_0 < t_1 < \dots < t_I = t_f.$$

We define $h_i = t_{i+1} - t_i$. The state y is approximated by a piecewise polynomial $y^{h,N}$. The restriction of $y^{h,N}$ onto $[t_i, t_{i+1}]$, $i = 0, \dots, I-1$, is denoted by $y_i^{h,N}$ and written as

$$y_i^{h,N}(t) = \sum_{j=0}^N y_{i,j} \psi_j \left(-1 + 2 \frac{t - t_i}{h_i} \right), \quad (3.70)$$

The collocation discretization of the optimal control problem (2.5) is given by

$$\min \quad m(y_{I-1,N}), \quad (3.71a)$$

s.t.

$$\mathbf{D} \begin{pmatrix} y_{i,0} \\ \vdots \\ y_{i,N} \end{pmatrix} = \frac{h_i}{2} \begin{pmatrix} f(y_{i,0}, u_{i,0}) \\ \vdots \\ f(y_{i,N}, u_{i,N}) \end{pmatrix}, \quad i = 0, \dots, I-1, \quad (3.71b)$$

$$y_{i,N} = y_{i+1,0}, \quad i = 0, \dots, I-2, \quad (3.71c)$$

$$y_{0,0} = \bar{y}_0, \quad (3.71d)$$

$$b(y_{I-1,N}) = 0. \quad (3.71e)$$

The weighted Lagrangian corresponding to (3.71) is given by

$$L_w(\mathbf{y}, \mathbf{u}, \boldsymbol{\lambda}, \bar{\boldsymbol{\mu}}, \mu_0, \mu_N) = m(y_{I-1,N}) \quad (3.72)$$

$$\begin{aligned} & + \sum_{i=0}^{I-1} \sum_{j=0}^N w_j \lambda_{i,j}^T \left[\frac{h_i}{2} f(y_{i,j}, u_{i,j}) - \sum_{k=0}^N D_{j,k} y_{i,k} \right] \\ & + \sum_{i=0}^{I-2} \bar{\mu}_i^T [y_{i,N} - y_{i+1,0}] \\ & + b(y_{I-1,N})^T \mu_N + (y_{0,0} - \bar{y}_0)^T \mu_0. \end{aligned} \quad (3.73)$$

Differentiating the Lagrangian (3.72) with respect to the $y_{i,j}$'s and setting the derivatives to zero gives the weighted-discrete adjoint equations on multiple intervals

$$\begin{aligned} \frac{h_0}{2} f_y(y_{0,0}, u_{0,0})^T w_0 \lambda_{0,0} - \sum_{k=0}^N D_{k,0} w_k \lambda_{0,k} + \mu_0, &= 0, \\ \frac{h_0}{2} f_y(y_{0,j}, u_{0,j})^T w_j \lambda_{0,j} - \sum_{k=0}^N D_{k,j} w_k \lambda_{0,k}, & j = 1, \dots, N-1, \\ \frac{h_0}{2} f_y(y_{0,N}, u_{0,N})^T w_N \lambda_{0,N} - \sum_{k=0}^N D_{k,N} w_k \lambda_{0,k} + \bar{\mu}_0, &= 0, \end{aligned} \quad (3.74a)$$

on the first subinterval $i = 0$,

$$\begin{aligned} \frac{h_i}{2} f_y(y_{i,0}, u_{i,0})^T w_0 \lambda_{i,0} - \sum_{k=0}^N D_{k,0} w_k \lambda_{i,k} - \bar{\mu}_{i-1}, &= 0, \\ \frac{h_i}{2} f_y(y_{i,j}, u_{i,j})^T w_j \lambda_{i,j} - \sum_{k=0}^N D_{k,j} w_k \lambda_{i,k}, & j = 1, \dots, N-1, \\ \frac{h_i}{2} f_y(y_{i,N}, u_{i,N})^T w_N \lambda_{i,N} - \sum_{k=0}^N D_{k,N} w_k \lambda_{i,k} + \bar{\mu}_i, &= 0, \end{aligned} \quad (3.74b)$$

for $i = 1, \dots, I-2$, and

$$\begin{aligned} \frac{h_i}{2} f_y(y_{I-1,0}, u_{I-1,0})^T w_0 \lambda_{I-1,0} - \sum_{k=0}^N D_{k,0} w_k \lambda_{I-1,k} - \bar{\mu}_{I-2}, &= 0, \\ \frac{h_i}{2} f_y(y_{I-1,j}, u_{I-1,j})^T w_j \lambda_{I-1,j} - \sum_{k=0}^N D_{k,j} w_k \lambda_{I-1,k}, & j = 1, \dots, N-1, \\ \frac{h_i}{2} f_y(y_{I-1,N}, u_{I-1,N})^T w_N \lambda_{I-1,N} - \sum_{k=0}^N D_{k,N} w_k \lambda_{I-1,k} \\ + b_y(y_{I-1,N})^T \mu_N + m_y(y_{I-1,N}), &= 0, \end{aligned} \quad (3.74c)$$

on the last subinterval $i = I-1$.

The next lemma shows that the adjoint variable p divided by the weighting function g satisfies the weighted-discrete adjoint equations along multiple subintervals (3.19a) with an error that is dependent on the true adjoint p , the weighting function g and N .

Lemma 3.10 If p, q_0 and q_f satisfy the adjoint equation (2.6b) and the transversality conditions (2.6c) then

$$\begin{aligned} & \frac{h_0}{2} f_y(y_{0,0}, u_{0,0})^T \frac{p(t_{0,0})}{g(c_0)} - \sum_{k=0}^N D_{k,0} \frac{p(t_{0,k})}{g(c_k)} + q_0 \\ &= r_{0,0}^a(N, p, g), \\ & \frac{h_0}{2} f_y(y_{0,j}, u_{0,j})^T \frac{p(t_{0,j})}{g(c_j)} - \sum_{k=0}^N D_{k,j} \frac{p(t_{0,k})}{g(c_k)} \\ &= r_{0,j}^a(N, p, g), \quad j = 1, \dots, N-1, \\ & \frac{h_0}{2} f_y(y_{0,N}, u_{0,N})^T \frac{p(t_{0,N})}{g(c_N)} - \sum_{k=0}^N D_{k,N} \frac{p(t_{0,k})}{g(c_k)} + p(t_{0,N}) \\ &= r_{0,N}^a(N, p, g), \end{aligned} \tag{3.75a}$$

on the first subinterval $i = 0$,

$$\begin{aligned} & \frac{h_i}{2} f_y(y_{i,0}, u_{i,0})^T \frac{p(t_{i,0})}{g(c_0)} - \sum_{k=0}^N D_{k,0} \frac{p(t_{i,k})}{g(c_k)} - p(t_{i-1,N}) \\ &= r_{i,0}^a(N, p, g), \\ & \frac{h_i}{2} f_y(y_{i,j}, u_{i,j})^T \frac{p(t_{i,j})}{g(c_j)} - \sum_{k=0}^N D_{k,j} \frac{p(t_{i,k})}{g(c_k)} \\ &= r_{i,j}^a(N, p, g), \quad j = 1, \dots, N-1, \\ & \frac{h_i}{2} f_y(y_{i,N}, u_{i,N})^T \frac{p(t_{i,N})}{g(c_N)} - \sum_{k=0}^N D_{k,N} \frac{p(t_{i,k})}{g(c_k)} + p(t_{i,N}) \\ &= r_{i,N}^a(N, p, g), \end{aligned} \tag{3.75b}$$

for $i = 1, \dots, I - 2$, and

$$\begin{aligned}
& \frac{h_i}{2} f_y(y_{I-1,0}, u_{I-1,0})^T \frac{p(t_{I-1,0})}{g(c_0)} - \sum_{k=0}^N D_{k,0} \frac{p(t_{I-1,k})}{g(c_k)} - p(t_{I-2,N}) \\
&= r_{I-1,0}^a(N, p, g), \\
& \frac{h_i}{2} f_y(y_{I-1,j}, u_{I-1,j})^T \frac{p(t_{I-1,j})}{g(c_j)} - \sum_{k=0}^N D_{k,j} \frac{p(t_{I-1,k})}{g(c_k)} \\
&= r_{I-1,j}^a(N, p, g), \quad j = 1, \dots, N - 1, \\
& \frac{h_i}{2} f_y(y_{I-1,N}, u_{I-1,N})^T \frac{p(t_{I-1,N})}{g(c_N)} - \sum_{k=0}^N D_{k,N} \frac{p(t_{I-1,k})}{g(c_k)} \\
&+ b_y(y_{I-1,N})^T q_f + m_y(y_{I-1,N}) \\
&= r_{I-1,N}^a(N, p, g),
\end{aligned} \tag{3.75c}$$

on the last subinterval $i = I - 1$, where

$$r_{i,j}^a(N, p, g) = \frac{w_j}{g(c_j)} \left[\frac{d}{dt} P_N(p)(t_{i,j}) - \frac{d}{dt} p(t_{i,j}) \right] + \epsilon_{i,j}(N, p, g), \tag{3.76}$$

with

$$\begin{aligned}
\epsilon_{i,j}(N, p, g) &= \int_{-1}^1 g(t) \left(P_N \left(\frac{p(t_i + (h_i/2) \cdot)}{g} \right)(t) - \frac{p(t_i + (h_i/2)t)}{g(t)} \right) \frac{d}{dt} \psi_j(t) dt \\
&+ \int_{-1}^1 g(t) \left(P_N \left(\frac{\psi_j}{g} \right)(t) - \frac{\psi_j(t)}{g(t)} \right) \frac{d}{dt} P_N(p(t_i + (h_i/2) \cdot))(t) dt \\
&+ \int_{-1}^1 g(t) \frac{\psi_j(t)}{g(t)} \frac{d}{dt} (P_N(p(t_i + (h_i/2) \cdot))(t) - p(t_i + (h_i/2)t)) dt.
\end{aligned} \tag{3.77}$$

Proof The result is a direct extension of Lemma 3.2. It is obtained by using equation (3.23) and the fact that p, q_0 and q_f satisfy the adjoint equation (2.6b) and the transversality conditions (2.6c). Then $p/g, p, q_0$ and q_f are inserted into (3.74) for $\lambda, \bar{\mu}, \mu_0$ and μ_N respectively to obtain (3.75). \square

Chapter 4

International Space Station Momentum Dumping Problem

The Legendre Pseudospectral method described in the previous chapters is now applied to a realistic optimal control problem. This chapter describes formulation and solution of the International Space Station momentum dumping problem. One version of this problem, where a continuous control is considered, lends itself to the application of the Legendre Pseudospectral method on one interval while other versions, where piecewise constant controls are considered, lend themselves to the Legendre Pseudospectral method using multiple subintervals. In each case, the problem is stated, then transcribed into a nonlinear program and solved using standard nonlinear programming techniques. Numerical results for each problem scenario are given.

4.1 Background

Spacecraft attitude control is usually provided by momentum devices such Control Moment Gyroscopes (CMGs) or reaction wheels, as they do not require consumables. However, the momentum of such devices is limited and when this limit is reached the device is termed saturated. In this situation, ‘controllability’ is lost along the momentum saturation direction. Recovering full three degree-of-freedom control requires desaturating the momentum device.

The usual approach to desaturate accumulated momentum is to use an additional device. Examples are mass expulsion devices, magnetic dipoles which interact with the Earth’s magnetic field, and rotating solar arrays which interact with solar radiation pressure [22],[50]. Mass expulsion devices require the use of consumable

propellant, which has finite lifetime and is expensive to get to orbit or replenish. Magnetic dipoles are electromagnets, which generate a torque on the spacecraft by their dipole interaction with the Earth's magnetic field. Disadvantages are that the Earth's magnetic field is not well known and hence may require the use of magnetometers to measure it, that it can be affected by sun spots or magnetic storms, and that it varies with orbit location thus restricting the amount and direction in which momentum can be unloaded. Roll and pitch momentum is unloaded near the magnetic poles, while roll and yaw momentum is unloaded near the geomagnetic equator [32],[51]. Further, the use of dipoles generates an additional magnetic field on the spacecraft, which may affect other sensors or devices. Solar pressure based methods require the use of modulating surfaces such as solar arrays. This sacrifices electrical power, creates mechanical lifetime issues due to wear and tear and increases the risk of drive failure.

An alternative is to use the momentum devices to appropriately maneuver the spacecraft in a disturbance field such that accumulated momentum can be removed [51]. Since most environmental disturbances on the spacecraft are a function of its attitude, the accumulated momentum due to navigating in such a disturbance field is path dependent. Performing an attitude maneuver over a pre-selected trajectory can result in a lower final momentum state than one with which the vehicle started. The advantage of this approach is that it does not require any additional hardware or specialized software. Hence, it can be applied to any existing vehicles that use momentum devices. In general, the method provides momentum unloading in all axes and does not require preferred orbit locations. Gravity gradient and to a lesser extent aerodynamic torques are well defined and better known than Earth's magnetic field. This approach can also be used as a contingency operational mode for spacecraft which use other actuators for momentum dumping purposes, consequently increasing the operational lifetime of satellites already in orbit.

4.2 Rotational Dynamics

The equations of motion described in this section can be found in [32]. The attitude dynamics of a rigid body in a circular orbit are given as

$$J \frac{d}{dt} \omega(t) = \tau_d - \omega(t)^\times (J\omega(t) + h(t)) - u(t), \quad (4.1a)$$

where $\omega : \mathbb{R} \mapsto \mathbb{R}^3$ is the angular velocity of the spacecraft with respect to an inertial reference frame measured in rad/sec. The remaining terms are $h : \mathbb{R} \mapsto \mathbb{R}^3$ the angular momentum of the Control Moment Gyroscopes (CMGs) measured in ft-lbs-sec, $u : \mathbb{R} \mapsto \mathbb{R}^3$ the control torque measured in ft-lbs, $\tau_d \in \mathbb{R}^3$ the external disturbance torque, $J \in \mathbb{R}^{3 \times 3}$ the inertia matrix of the spacecraft measured in slugs-ft². All terms are evaluated with respect to the spacecrafts fixed body reference frame, see Figure 4.1. The skew-symmetric cross product operator $^\times$ is given as

$$\omega^\times = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}.$$

External disturbance torques may take many forms. Examples are gravity gradient torque, aerodynamic torque and magnetic torque. This model considers only gravity gradient torque τ_{gg} which is given by

$$\tau_{gg} = 3\omega_{orb}^2 C_3^\times J C_3, \quad (4.1b)$$

where $\omega_{orb} = 0.0011$ rad/sec is the orbital for the current altitude and C_3 is the third column of the rotation matrix which rotates any vector in the local vertical local horizontal (LVLH) reference frame into the spacecrafts body reference frame. It is assumed that all other external torques are small relative to those modeled and therefore negligible.

The attitude kinematics, using a quaternion formulation, are given as

$$\frac{d}{dt} q(t) = T(q)(\omega(t) - \omega_o), \quad (4.1c)$$

where $q : \mathbb{R} \mapsto \mathbb{R}^4$ is the attitude quaternion, $T : \mathbb{R}^4 \mapsto \mathbb{R}^{4 \times 3}$ is given by

$$T(q) = \frac{1}{2} \begin{pmatrix} -q_2(t) & -q_3(t) & -q_4(t) \\ q_1(t) & -q_4(t) & q_3(t) \\ q_4(t) & q_1(t) & -q_2(t) \\ -q_3(t) & q_2(t) & q_1(t) \end{pmatrix}$$

and $\omega_o \in \mathbb{R}^3$ is the constant orbital rate for the LVLH reference frame, assuming a circular orbit. The $\omega(t) - \omega_o$ term represents the relative angular rate with respect to the LVLH reference frame, therefore the quaternion q computed from (4.1c) describes the attitude of the spacecraft with respect to the LVLH reference frame. It is standard for the control variable to enter into the dynamics through a control law. For now it is assumed that the CMGs are controllable directly, resulting in the control law

$$\frac{d}{dt} h(t) = u(t). \quad (4.1d)$$

Figure (4.1) shows the aforementioned reference frames as they relate to the Earth, the LVLH orbit and the space station.

Additional consideration must be given to the attitude kinematics equation (4.1c) because a quaternion must always have a unit norm. Therefore a path equality constraint must be added

$$\|q(t)\|_2 = 1, \quad \forall t \in [t_0, t_f]. \quad (4.2)$$

This is typically a difficult constraint to satisfy. For simulations, the standard procedure in simulations is to divide the current quaternion by its magnitude during each step of numerical integration. In this thesis we use Legendre Pseudospectral collocation to discretize the dynamics in the optimal control problem. Since equality constraints are only enforced at the collocation points, it should not be expected that the unity norm constraint will be satisfied in the infinite dimensional sense. Such a constraint violation results in a solution which has no physical meaning. Therefore it is beneficial to use Euler-Rodriguez parameters which are defined to be [32]

$$r(t) = \frac{1}{q_1(t)} (q_2(t) \ q_3(t) \ q_4(t))^T. \quad (4.3)$$

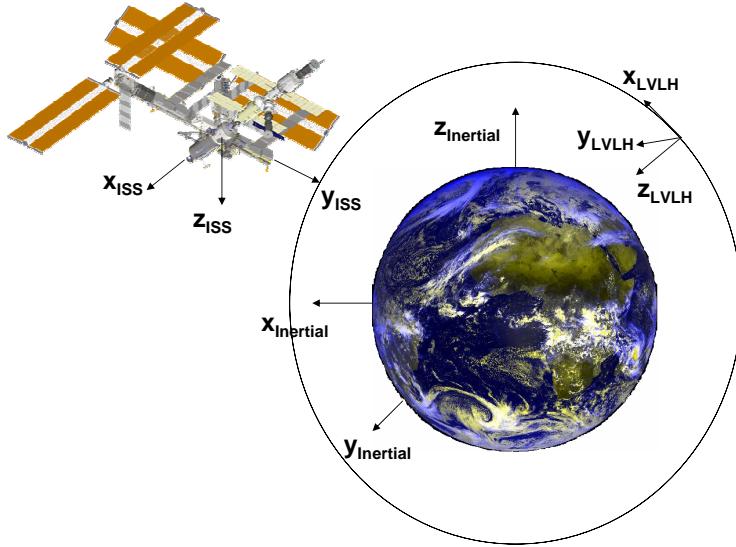


Figure 4.1: Earth's Inertial Reference Frame, Local Vertical Local Horizontal Reference Frame, International Space Station's Body Reference Frame (ISS Assembly 12A shown)

Note that (4.3) is not defined when $q_1(t) = 0$, which is equivalent to a 180° rotation. This corresponds to attitudes which are assumed to not occur along the optimal trajectory. Using the representation (4.3) does not require the path equality constraint (4.2). Via Euler-Rodriguez parameters, (4.1c) can be converted to

$$\frac{d}{dt}r(t) = \frac{1}{2}(r(t)r(t)^T + I + r(t)^\times)(\omega(t) - \omega_o). \quad (4.4)$$

The rotation matrix in (4.1b) can be computed as

$$C = I + \frac{2}{1 + r^T r}(r^\times r^\times - r^\times).$$

The resulting attitude dynamics for the space station are given as

$$\begin{aligned} J \frac{d}{dt} \omega(t) &= \tau_{gg}(r) - \omega(t)^\times (J\omega(t) + h(t)) - u(t) \\ \frac{d}{dt} r(t) &= \frac{1}{2}(r(t)r(t)^T + I + r(t)^\times)(\omega(t) - \omega_o(r)) \\ \frac{d}{dt} h(t) &= u(t). \end{aligned} \quad (4.5)$$

4.2.1 International Space Station Assembly Stage 12A

This thesis considers International Space Station stage 12A, which was originally scheduled for launch in December 2002. This assembly makes the following additions [2]:

- Delivers second port truss segment (P3/P4 truss) to attach to first port truss segment (P1 truss).
- Central cooling radiators, delivered earlier on flights 9A and 11A, are deployed from first starboard (S1 truss) port (P1) truss segments.
- Exterior attachments for Brazilian Unpressurized Logistics Carriers (ULCs) are delivered.

The inertia matrix J for space station assembly stage 12A, shown in Figure 4.2, is given in Table 4.1 [40].

2.8070×10^7	4.8225×10^5	-1.7168×10^7
4.8225×10^5	9.5145×10^7	6.0260×10^4
-1.7168×10^7	6.0260×10^4	7.6594×10^7

Table 4.1: ISS 12A Inertia Matrix [slugs-ft²]

Due to physical limitations, the CMGs must not reach a certain momentum magnitude threshold because they will become saturated. This saturation limit can be found in [39] to be 10000 ft-lbs-sec. This leads to the path inequality constraint

$$\|h(t)\|_2 \leq h_{max}, \quad \forall t \in [t_0, t_f], \quad (4.6)$$

where $h_{max} = 10000$.

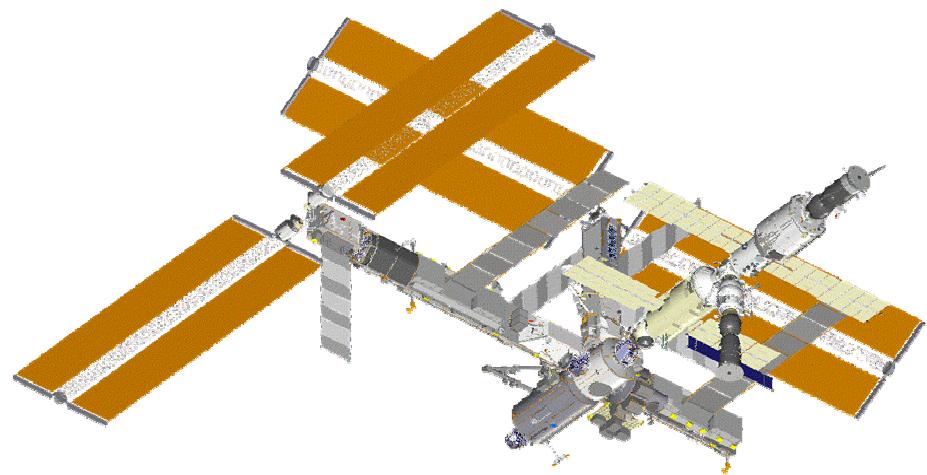


Figure 4.2: International Space Station Assembly 12A

4.2.2 Boundary Conditions

The boundary conditions for (4.5) are chosen such that the spacecraft is at a Principal Axis (PA) attitude initially and travels to a Torque Equilibrium Attitude (TEA). A PA attitude is an attitude for which the gravity gradient torque is zero. A PA is a common rest attitude for a spacecraft such as the space station. The PA attitude associated with the inertia matrix, in Table 4.1, is shown in Table 4.2.

$\omega(t_0)$	-9.54×10^{-6}	-1.14×10^{-3}	5.35×10^{-6}
$r(t_0)$	3.00×10^{-3}	1.53×10^{-1}	3.83×10^{-3}

Table 4.2: Attitude and Rate Corresponding to Principal Axis

A TEA is a special attitude for which the right-hand side of the differential equations (4.5) are all zero when no control is exerted in the vehicle. Finding and reaching a TEA reduces to a final time boundary condition where a root-finding problem to find ω, r and h such that the right hand side of the differential equation is equal to zero when $u = 0$. A TEA corresponds to attitudes that can be held indefinitely. This is a desirable attitude because when a spacecraft is at a TEA, it does not require attitude control devices to stay at that attitude. In other words, the final state of the system can be maintained indefinitely with zero control effort.

Additionally, an initial value for angular momentum must be specified. This value can change from one simulation scenario to the next, so this condition is somewhat less strict as the PA and TEA requirements. For problems considered in this thesis, the initial value

$$h(t_0) = (5000, 5000, 5000)^T \text{ ft-lbs-sec}$$

was used.

These boundary conditions can be written compactly as

$$\omega(t_0) = \bar{\omega}_0,$$

$$r(t_0) = \bar{r}_0,$$

$$h(t_0) = \bar{h}_0,$$

and

$$b(\omega(t_f), r(t_f), h(t_f)) = \begin{pmatrix} J^{-1}(\tau_{gg}(t_f) - \omega(t_f)^\times(J\omega(t_f) + h(t_f))) \\ \frac{1}{2}(r(t_f)r(t_f)^T + I + r(t_f)^\times)(\omega(t_f) - \omega_o(r)) \end{pmatrix} = 0, \quad (4.7)$$

where $\bar{\omega}_0$, \bar{r}_0 and \bar{h}_0 are given by the above initial conditions.

4.3 ISS Momentum Dumping Problem with Continuous Control

From a modeling perspective, the simplest version of the space station momentum dumping problem is posed such that the control in (4.5) is a continuous function on the interval $[t_0, t_f]$. The optimal control problem can be stated as

$$\begin{aligned}
 \min \quad & ||h(t_f)||_2 \\
 \text{s.t.} \quad & \\
 J \frac{d}{dt} \omega(t) &= \tau_{gg}(r) - \omega(t)^\times (J\omega(t) + h(t)) - u(t), \quad t \in [t_0, t_f], \\
 \frac{d}{dt} r(t) &= \frac{1}{2}(r(t)r(t)^T + I + r(t)^\times)(\omega(t) - \omega_o(r)), \quad t \in [t_0, t_f], \\
 \frac{d}{dt} h(t) &= u(t), \quad t \in [t_0, t_f], \\
 ||h(t)||_2 &\leq h_{max}, \quad t \in [t_0, t_f], \\
 \omega(t_0) &= \bar{\omega}_0, \\
 r(t_0) &= \bar{r}_0, \\
 h(t_0) &= \bar{h}_0, \\
 b(\omega(t_f), r(t_f), h(t_f)) &= 0,
 \end{aligned} \tag{4.8}$$

where $b, \bar{\omega}_0, \bar{r}_0$ and \bar{h}_0 are given by (4.7). This problem is posed on the interval $[t_0, t_f] = [0, 1800]$ sec [38]. The initial data $\bar{\omega}_0, \bar{r}_0$ for the attitude and the angular rate were chosen to be the principal axis from Table 4.2. The initial value for the angular momentum was chosen to be $\bar{h}_0 = (5000, 5000, 5000)^T$, where $||\bar{h}_0||_2$ is close to h_{max} to make a desaturation maneuver meaningful. The final time boundary condition is defined in (4.7).

Since the control is continuous on the interval $[t_0, t_f]$ the Legendre Pseudospectral method with one time interval can be applied to discretize this problem. Applying

this direct transcription to (4.8) results in the following NLP

$$\begin{aligned}
\min \quad & \|h_N\|_2 \\
s.t. \quad & \\
& \frac{2}{t_f - t_0} J \sum_{k=0}^N D_{jk} \omega_k = \tau_{gg}(r_j) - \omega_j^\times (J\omega_j + h_j) - u_j, \quad j = 0, \dots, N, \\
& \frac{2}{t_f - t_0} \sum_{k=0}^N D_{jk} r_k = \frac{1}{2} (r_j r_j^T + I + r_j^\times) (\omega_j - \omega_o(r_j)), \quad j = 0, \dots, N, \\
& \frac{2}{t_f - t_0} \sum_{k=0}^N D_{jk} h_k = u_j, \quad j = 0, \dots, N, \\
& \|h_j\|_2 \leq h_{max}, \quad j = 0, \dots, N, \\
& \omega_0 = \bar{\omega}_0, \\
& r_0 = \bar{r}_0, \\
& h_0 = \bar{h}_0, \\
& b(\omega_N, r_N, h_N) = 0,
\end{aligned} \tag{4.9}$$

where the optimization variables are ω_j, r_j, h_j , and u_j , $j = 0, \dots, N$. Note that the path inequality constraint is only enforced at the collocation points. It is assumed that doing so will result in solutions that satisfy this constraint on the entire interval. The optimization problem (4.9) was solved with $N = 50$ using DIDO version 2003a [19], a MATLAB [37] based tool which implements the Legendre Pseudospectral collocation method, and uses SNOPT [24] to solve the resulting nonlinear program. MATLAB code for solving this problem can be found in Section B.1. Note that scaling factors were used to scale each variable in the optimization problem such that the constraint evaluations and variable magnitudes are of similar orders of magnitude. Values for the scaling factors used to solve this problem can be found in Section B.1. The initial guess for the NLP solver was obtained by using a constant control $u(t) = (0, 0, 0)^T$ and integrating the differential equations (4.5) forward using ODE45 [37].

The computed optimal solutions are shown in Figures 4.3-4.8, along with simulation results. The simulation results were obtained by inputting the computed optimal control values into a simulation which implements (4.5) and uses MATLAB's ODE45 for numerical integration.

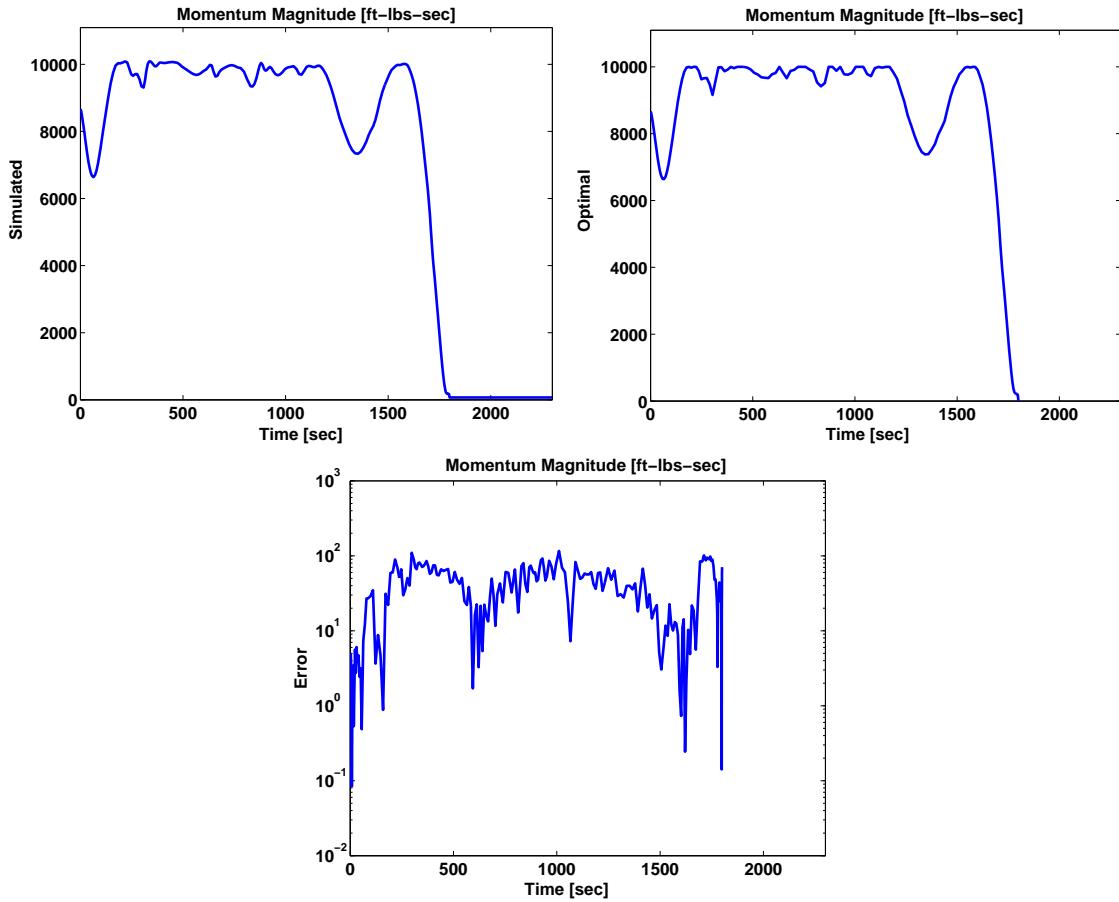


Figure 4.3: Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Continuous Control, Using N=150

Figure 4.3 shows the computed optimal objective function and the simulated objective function on the interval. It is apparent that the computed results and the simulated results are in close agreement, the CMG momentum magnitude was reduced from 8666 ft-lbs-sec to 0.1 ft-lbs-sec.

Figure 4.4 shows the computed optimal angular momentum values and the simulated angular momentum values on the interval. Just as for the magnitude, the angular momentum in each axis are in close agreement.

Figure 4.5 shows the optimal control computed by the Legendre Pseudospectral method. As one would expect, the control is very nonlinear which is due to the

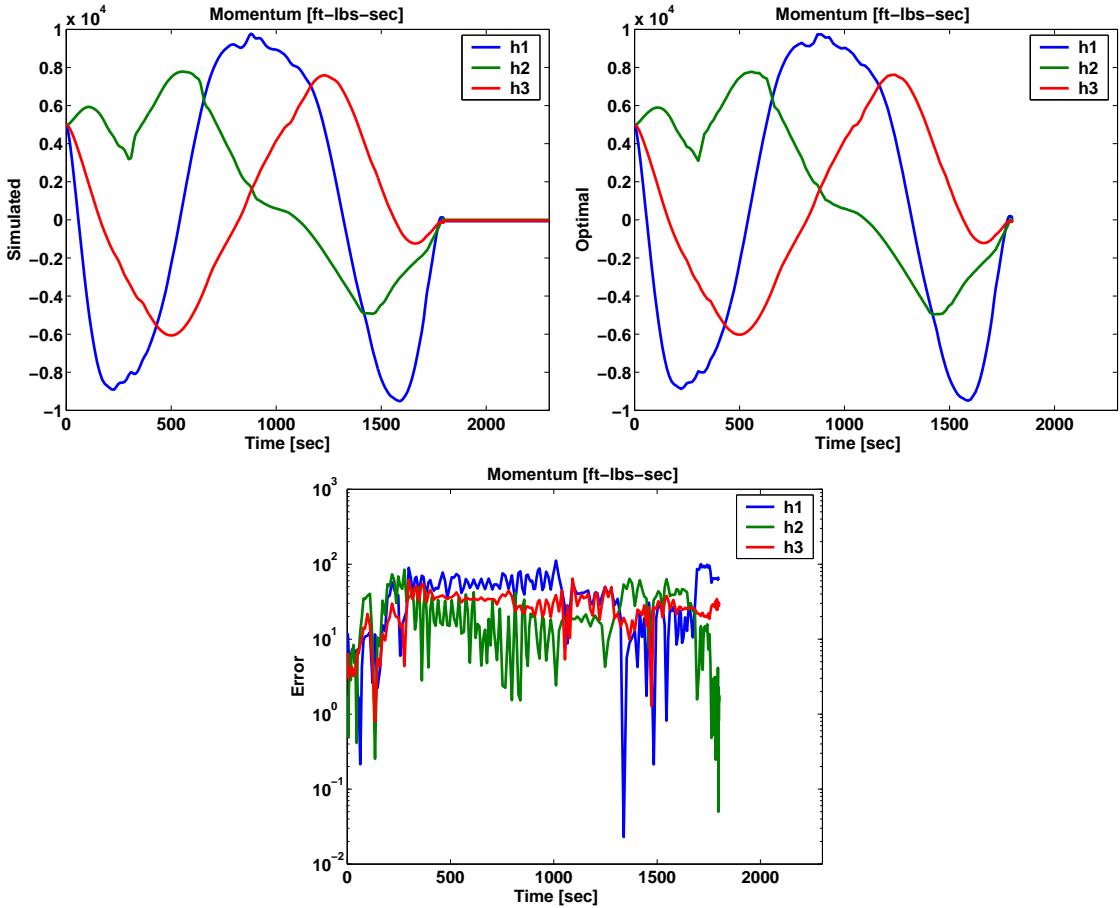


Figure 4.4: Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Continuous Control, Using N=150

nonlinear disturbances that are acting on the spacecraft. After the maneuver ended, at 1800 seconds, the control was set to zero to verify that a TEA was reached.

Figure 4.6 depicts the optimal attitude trajectory in Euler angles. The conversion from Euler-Rodrigues parameters to Euler angles can be found in [32]. As shown in [38] the optimal attitude trajectory for a desaturation maneuver resembles a sinusoid in the roll axis and the attitude trajectories for the pitch and yaw axes are fairly flat. The corresponding angular rate trajectories are shown in Figure 4.7. As indicated, after 1800 seconds the attitude and rate trajectories are constant. This verifies that a TEA was reached because no control was used to hold this attitude.

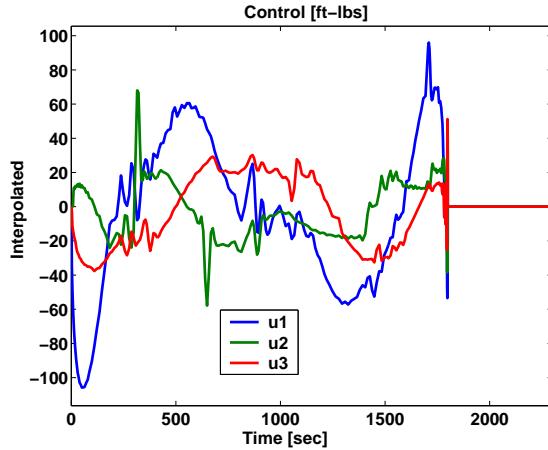


Figure 4.5: Computed Optimal Control, Extended by Zero for $t > 1800$, for the Space Station Momentum Dumping Problem with Continuous Control, Using $N=150$.

Figure 4.8 depicts the external torques that are acting on the spacecraft. As mentioned earlier, the nonlinearity in these torques account for the nonlinearity in the optimal control. As further proof that a TEA was reached, after the maneuver was completed these torques are either zero or counteract each other. This means that the right hand side of (4.1a) is zero. Figure 4.8 shows that the gravity gradient torque and the Euler torque combined were zero.

The optimal solution described in this section is meaningful in the sense that it solves the problem of interest. However this solution may not be directly implementable aboard the space station because computational storage limits and processing speed make it difficult to command a continuous control such as Figure 4.5. Obtaining implementable results requires that the optimal control problem be solved using piecewise constant controls.

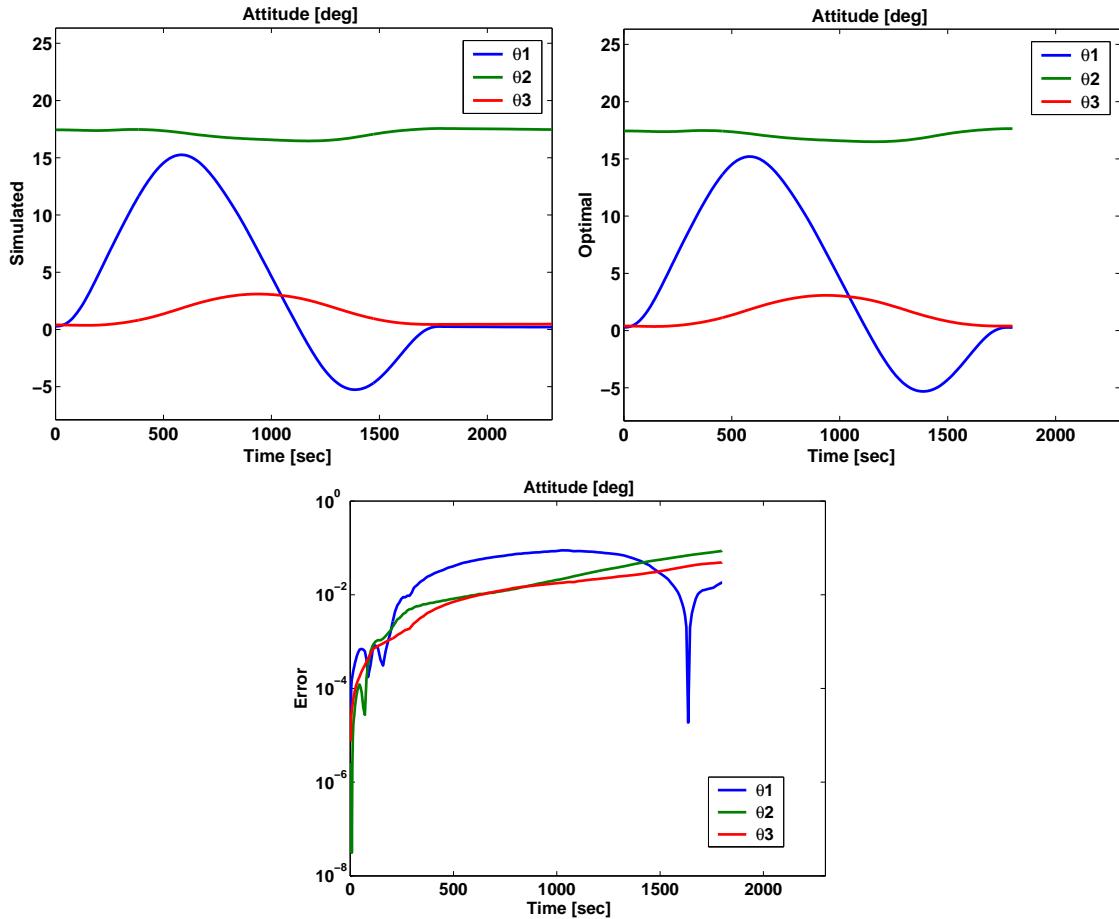


Figure 4.6: Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Continuous Control, Using N=150

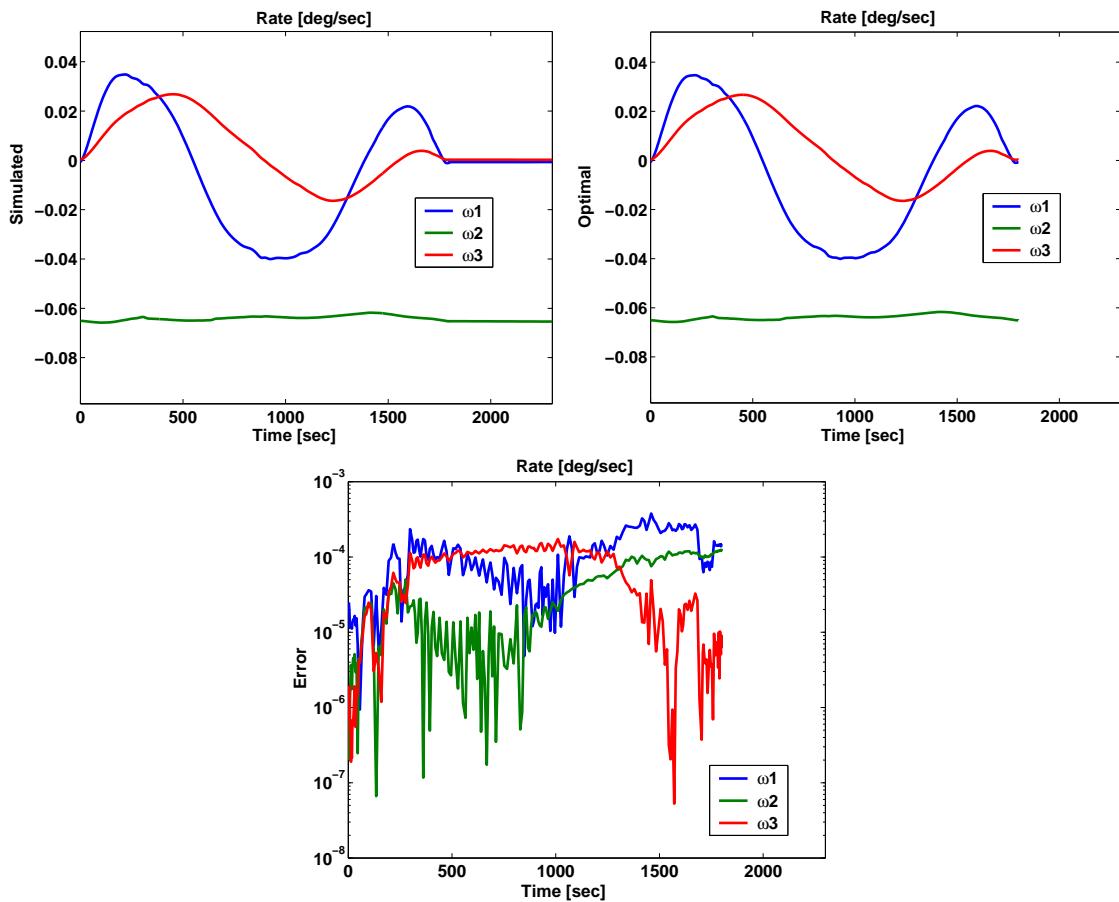


Figure 4.7: Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Continuous Control, Using N=150

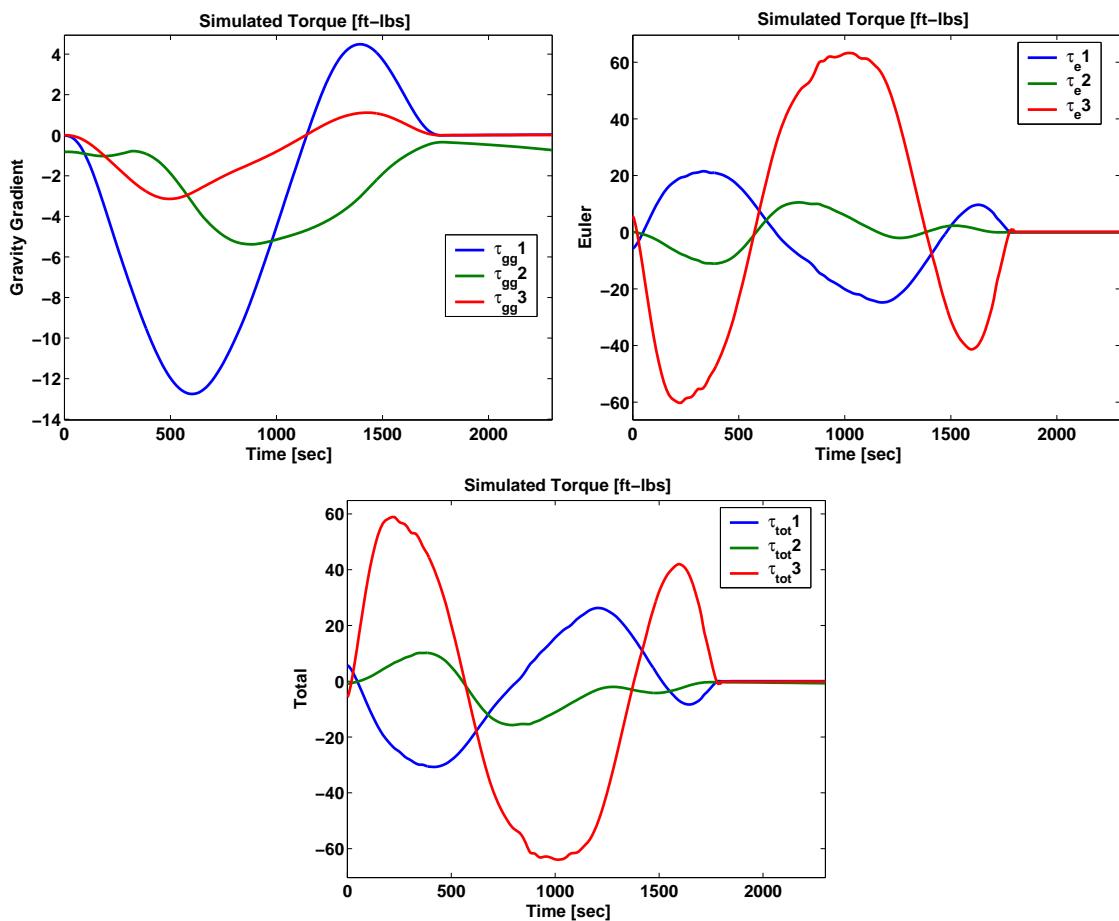


Figure 4.8: Simulated External Torques for Space Station Momentum Dumping Problem with Continuous Control, Using N=150

4.4 ISS Momentum Dumping Problem with Piecewise Constant Control

In contrast to Section 4.3, this version of the space station momentum dumping problem is posed such that the control in (4.5) is a piecewise constant function on the interval $[t_0, t_f]$. This is done because it is typical for the station to perform a series of constant attitude holds [39], which amount to having a piecewise constant control.

Again the optimal control problem is stated as

$$\begin{aligned}
 \min \quad & \|h(t_f)\|_2 \\
 \text{s.t.} \quad & \\
 J \frac{d}{dt} \omega(t) &= \tau_{gg}(r) - \omega(t)^\times (J\omega(t) + h(t)) - u(t), \quad t \in [t_0, t_f], \\
 \frac{d}{dt} r(t) &= \frac{1}{2}(r(t)r(t)^T + I + r(t)^\times)(\omega(t) - \omega_o(r)), \quad t \in [t_0, t_f], \\
 \frac{d}{dt} h(t) &= u(t), \quad t \in [t_0, t_f], \\
 \|h(t)\|_2 &\leq h_{max}, \quad t \in [t_0, t_f], \\
 \omega(t_0) &= \bar{\omega}_0, \\
 r(t_0) &= \bar{r}_0, \\
 h(t_0) &= \bar{h}_0, \\
 b(\omega(t_f), r(t_f), h(t_f)) &= 0,
 \end{aligned} \tag{4.10}$$

where $b, \bar{\omega}_0, \bar{r}_0$ and \bar{h}_0 are given by (4.7). Initial and final conditions for the angular rate and the attitude as well as the initial value for the angular momentum are identical to those in Section 4.3,

Since the control is piecewise constant on each subinterval the Legendre Pseudospectral method with multiple subintervals of time must be applied to discretize this problem. This problem is posed on $I = 5$ subintervals due to the computational storage restrictions of onboard computers. The intervals are written as $[t_0, t_1]$, $[t_1, t_2]$, $[t_2, t_3]$, $[t_3, t_4]$, and $[t_4, t_5]$, with $t_0 = 0, t_5 = 1800$. Applying the pseudospectral

direct transcription to (4.8) results in the NLP

$$\begin{aligned}
\min \quad & \|h_{4N}\|_2 \\
\text{s.t.} \quad & \\
& \frac{2}{t_i - t_{i-1}} J \sum_{k=0}^N D_{jk} \omega_{ik} = \tau_{gg}(r_{ij}) - \omega_{ij}^\times (J \omega_{ij} + h_{ij}) - u_i, \\
& \quad j = 0, \dots, N, i = 0, \dots, 4, \\
& \frac{2}{t_i - t_{i-1}} \sum_{j=0}^N D_{jk} r_{ik} = \frac{1}{2} (r_{ij} r_{ij}^T + I + r_{ij}^\times) (\omega_{ij} - \omega_o(r_{ij})) \\
& \quad j = 0, \dots, N, i = 0, \dots, 4, \\
& \frac{2}{t_i - t_{i-1}} \sum_{j=0}^N D_{jk} h_{ik} = u_i, \quad j = 0, \dots, N, i = 0, \dots, 4, \\
& \\
& \|h_{ij}\|_2 \leq h_{max}, \quad j = 0, \dots, N, i = 0, \dots, 4, \\
& \omega_{i-1,N} = \omega_{i0}, \quad i = 1, \dots, 4, \\
& h_{i-1,N} = h_{i0}, \quad i = 1, \dots, 4, \\
& r_{i-1,N} = r_{i0}, \quad i = 1, \dots, 4, \\
& \omega_{0,0} = \bar{\omega}_0 \\
& r_{0,0} = \bar{r}_0 \\
& h_{0,0} = \bar{h}_0 \\
& b(\omega_{4,N}, r_{4,N}, h_{4,N}) = 0,
\end{aligned} \tag{4.11}$$

where the optimization variables once again are $\omega_{ij}, r_{ij}, h_{ij}$ and u_i , $j = 0, \dots, N$, $i = 0, \dots, 4$. The problem (4.11) has four new optimization variables, t_1, t_2, t_3 and t_4 , because the times for which the control changes are also to be determined. As in (4.9) the path inequality constraint is only enforced at the collocation points and it is assumed that doing so will result in solutions that satisfy this constraint on the entire interval. The optimization problem (4.11) was solved with $N = 30$ on each subinterval using DIDO version 2003a which relies on SNOPT to solve the resulting nonlinear program. MATLAB code for solving this problem can be found in Section B.2. Note that scaling factors were used to scale each variable in the optimization problem such that the constraint evaluations and variable magnitudes are of similar orders of

magnitude. Values for the scaling factors used to solve this problem can be found in Section B.2.

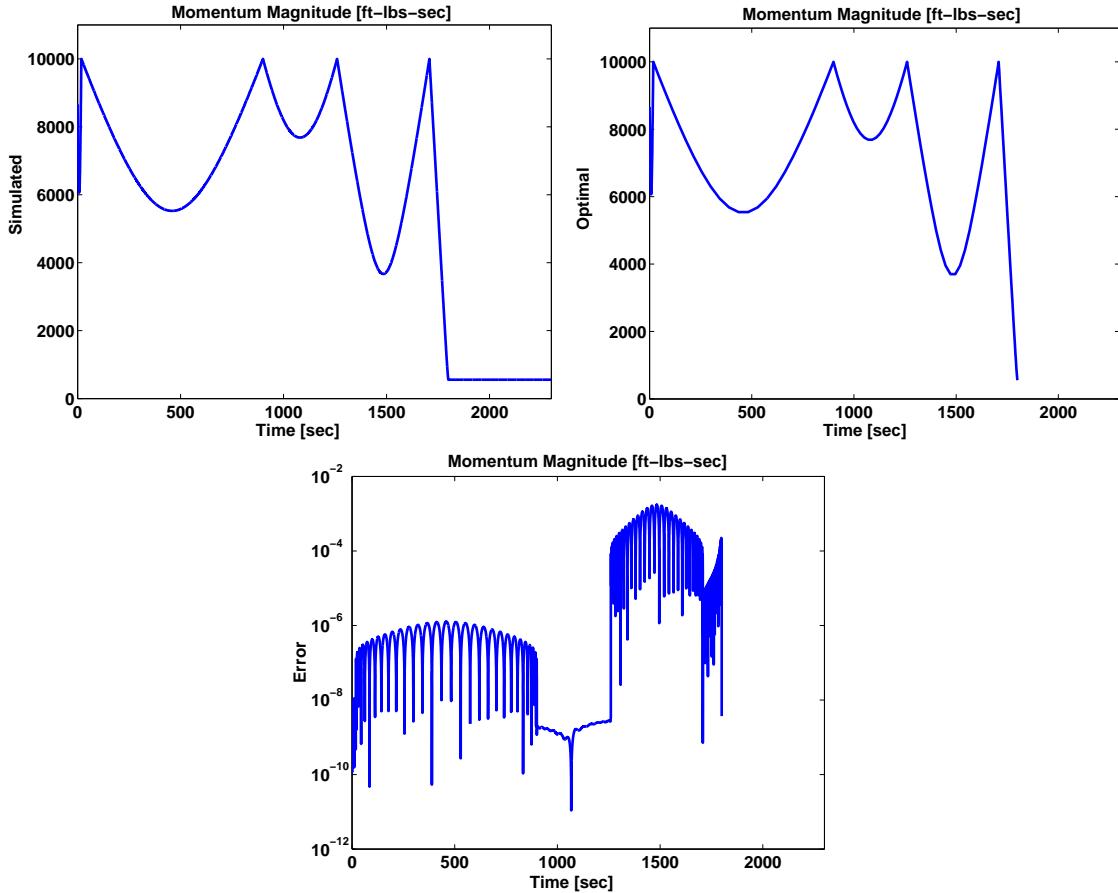


Figure 4.9: Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals

Figure 4.9 shows the computed optimal objective function and the simulated objective function on the interval. As indicated the computed results and the simulated results are in close agreement, despite the discontinuous control. The momentum magnitude was reduced by almost 6000 ft-lbs-sec, from 8666 ft-lbs-sec to 556 ft-lbs-sec. For this problem the objective function value is larger than the one computed in Section 4.3. Of course this is to be expected because the control is restricted to be from a smaller space of functions.

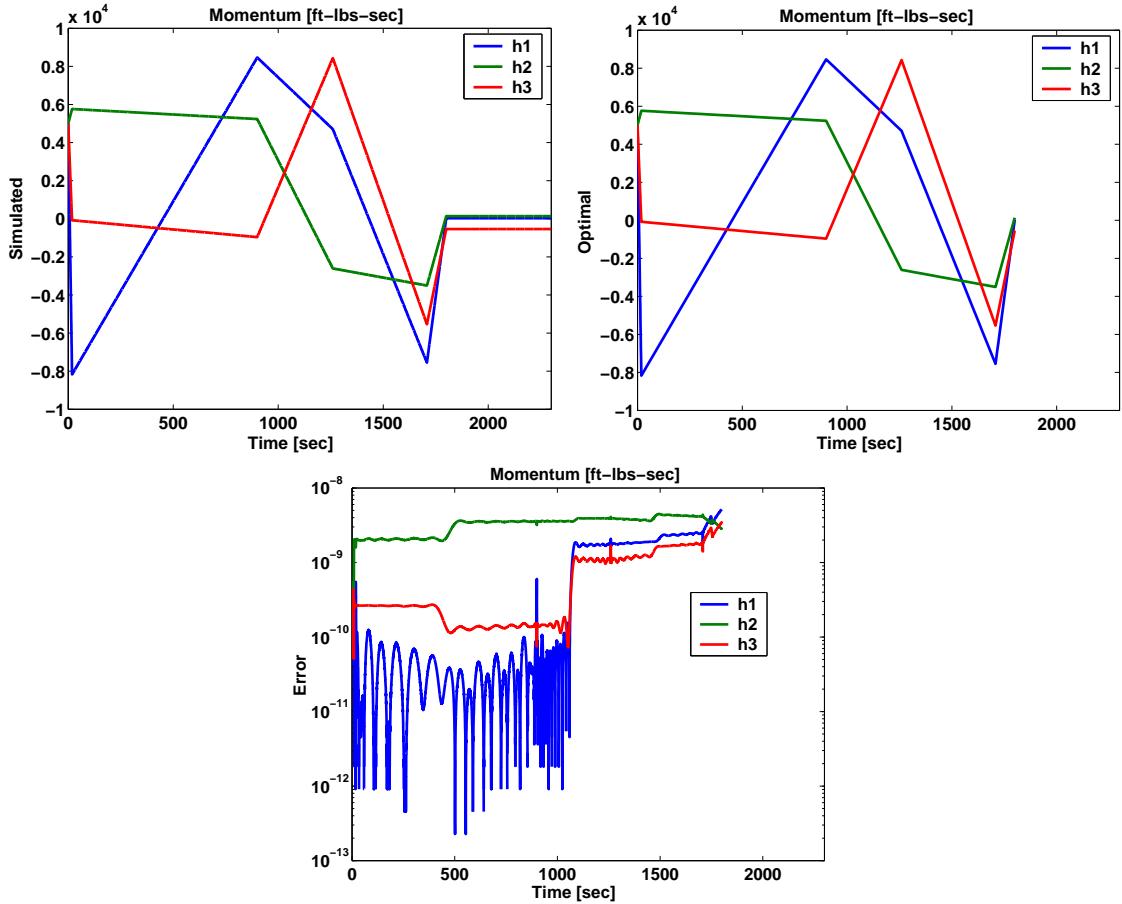


Figure 4.10: Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals

Figure 4.10 shows the computed optimal angular momentum values and the simulated angular momentum values on the interval. Just as for the magnitude, the angular momentum in each axis are in close agreement. Figure 4.11 shows the optimal control computed by the Legendre Pseudospectral method.

Figure 4.12 depicts the optimal attitude trajectory in Euler angles. The attitude trajectory is similar to the one in Section 4.3 but for kinks at the control transitions and the terminal values. As with previous results, the roll trajectory resembles a sinusoid while the pitch and yaw trajectories are fairly flat. The corresponding angular rate trajectories are shown in Figure 4.13. Due to the kinks in the attitude at

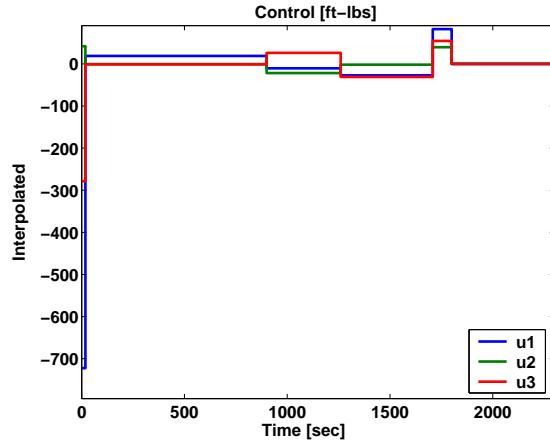


Figure 4.11: Computed Piecewise Constant Optimal Control, Extended by Zero for $t > 1800$, for Space Station Momentum Dumping Problem, Using $N=30$ on Five Subintervals.

the control transition points, the rate is nearly discontinuous at the points. Again, after 1800 seconds, the control was set to zero and the attitude and rate trajectories remained constant. This verifies that a TEA was reached because no control was used to hold this attitude.

Figure 4.14 depicts the external torques that are acting on the spacecraft. As further proof that a TEA was reached, after the maneuver was completed these torques are either zero or counteract each other. Figure 4.14 shows that the combined external torque goes to zero.

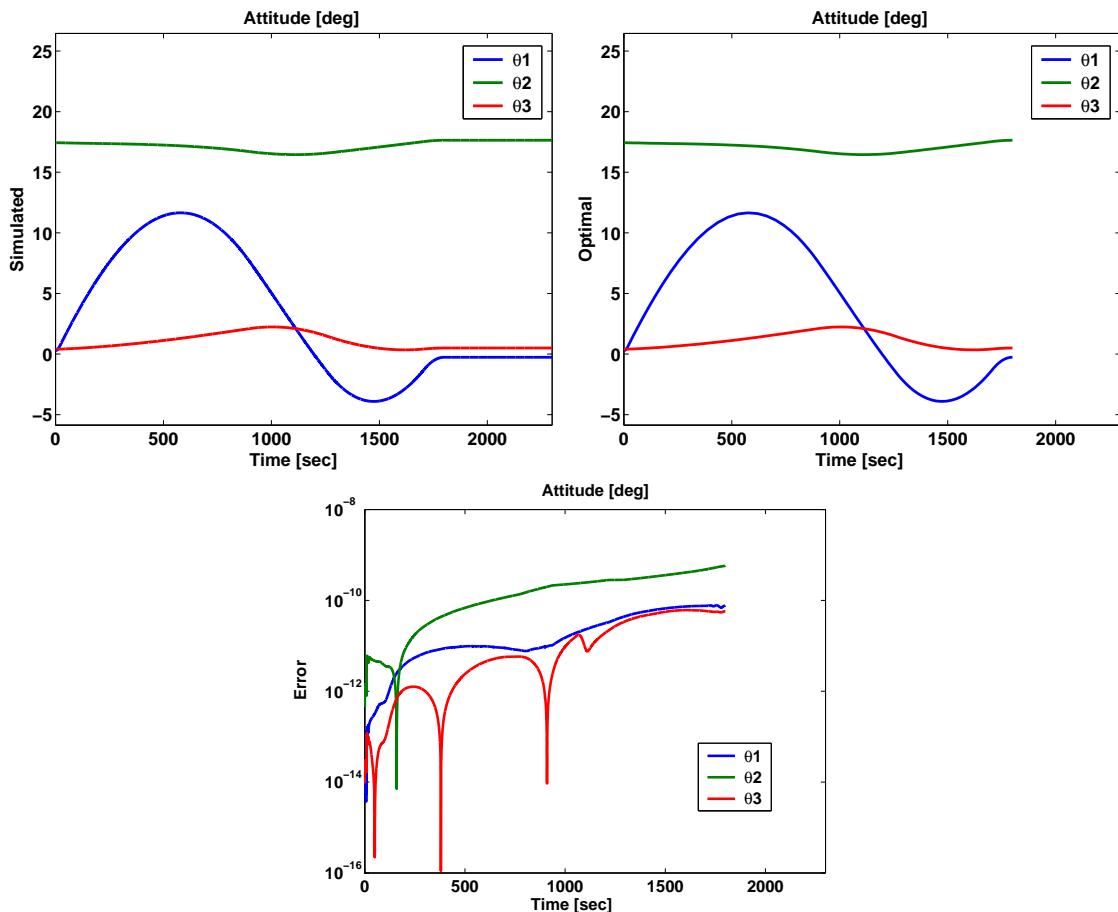


Figure 4.12: Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals

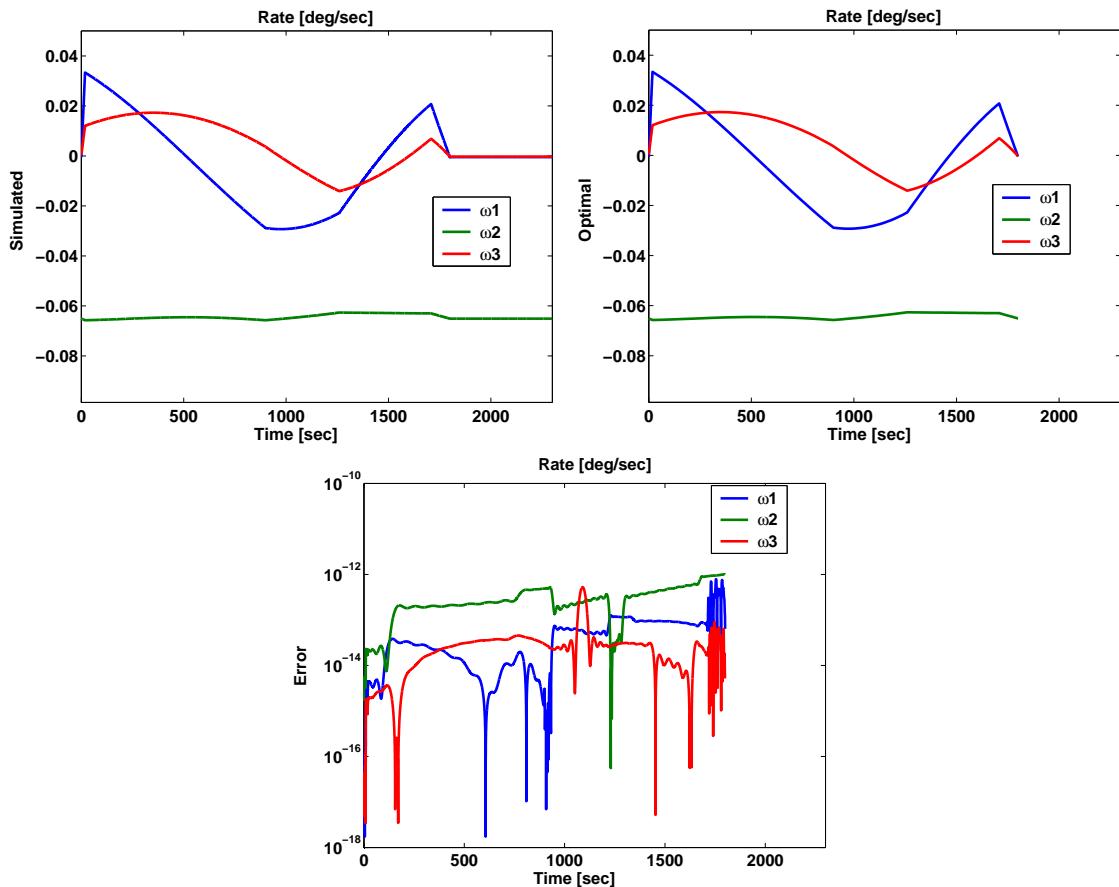


Figure 4.13: Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using $N=30$ on Five Subintervals

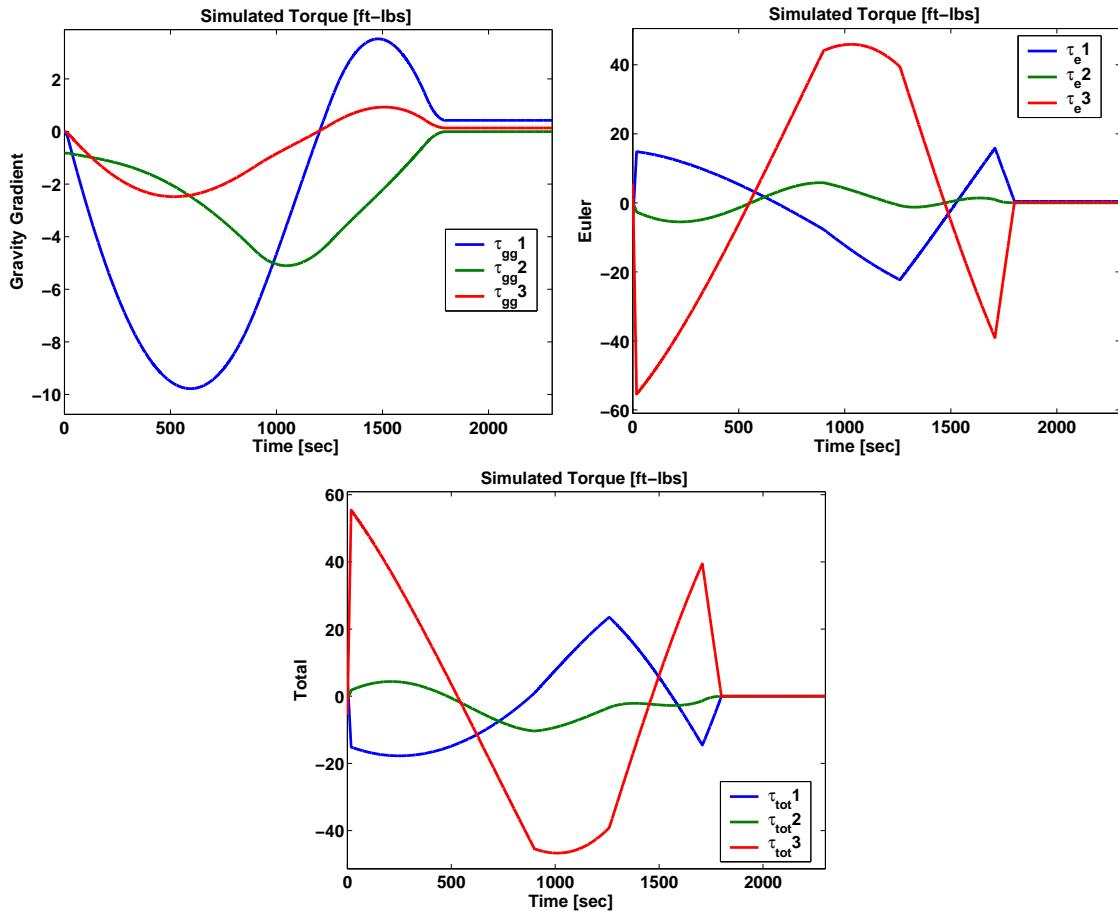


Figure 4.14: Simulated External Torques for Space Station Momentum Dumping Problem with Piecewise Constant Control, Using N=30 on Five Subintervals

4.4.1 Constraint on the Control Magnitude

Some control systems have a limit on the amount of torque that the attitude controller can generate at any given time. These limits manifest themselves in the form of a path inequality constraint on the control. Since the problem considered in Section 4.4 uses piecewise constant controls on each subinterval, constraining the control can be accomplished by simple bounds on the control on each subinterval. The constraint

$$\|u_i\|_2 \leq u_{max}, \quad i = 1, \dots, 5 \quad (4.12)$$

was added to the NLP (4.11) to produce the results shown in Figures 4.15 through 4.20 for $u_{max} = 200$ ft-lbs. MATLAB code for solving this problem can be found in Section B.2. Again, note that scaling factors were used to scale each variable in the optimization problem such that the constraint evaluations and variable magnitudes are of similar orders of magnitude. Values for the scaling factors used to solve this problem can be found in Section B.1.

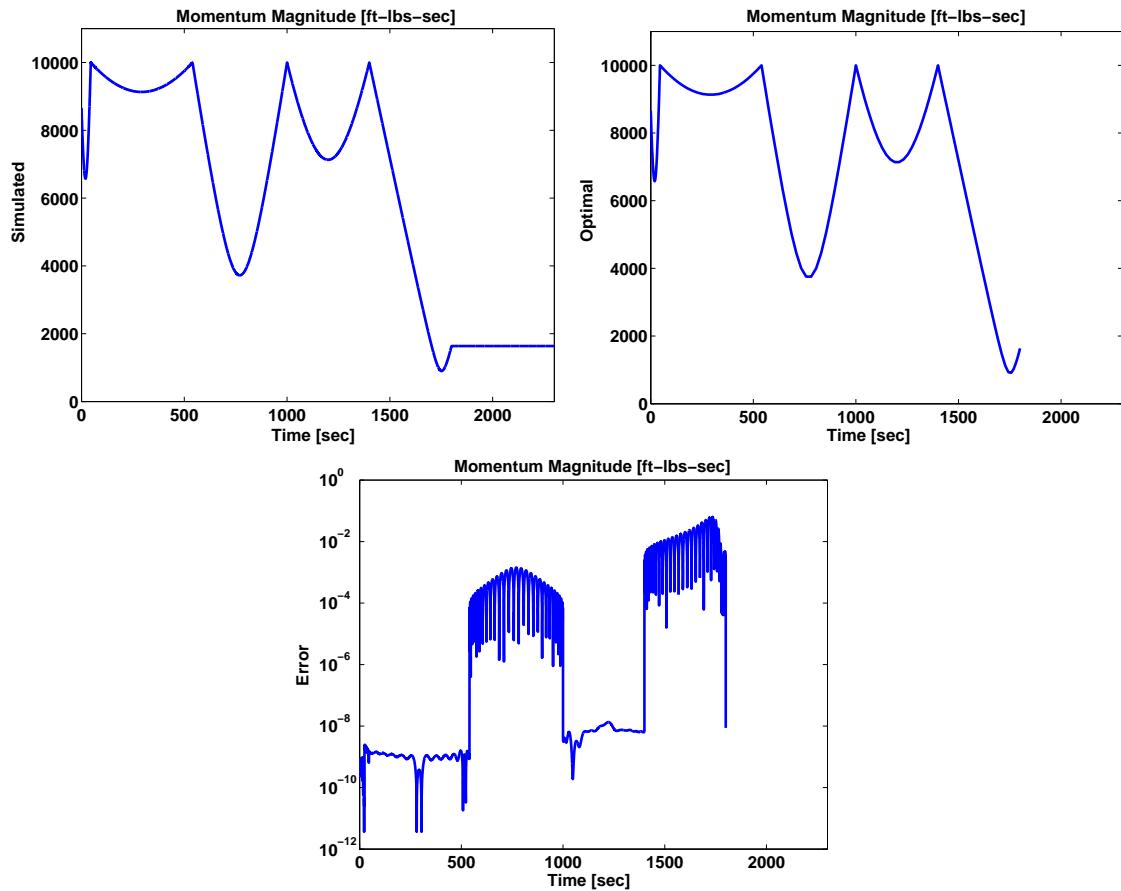


Figure 4.15: Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals

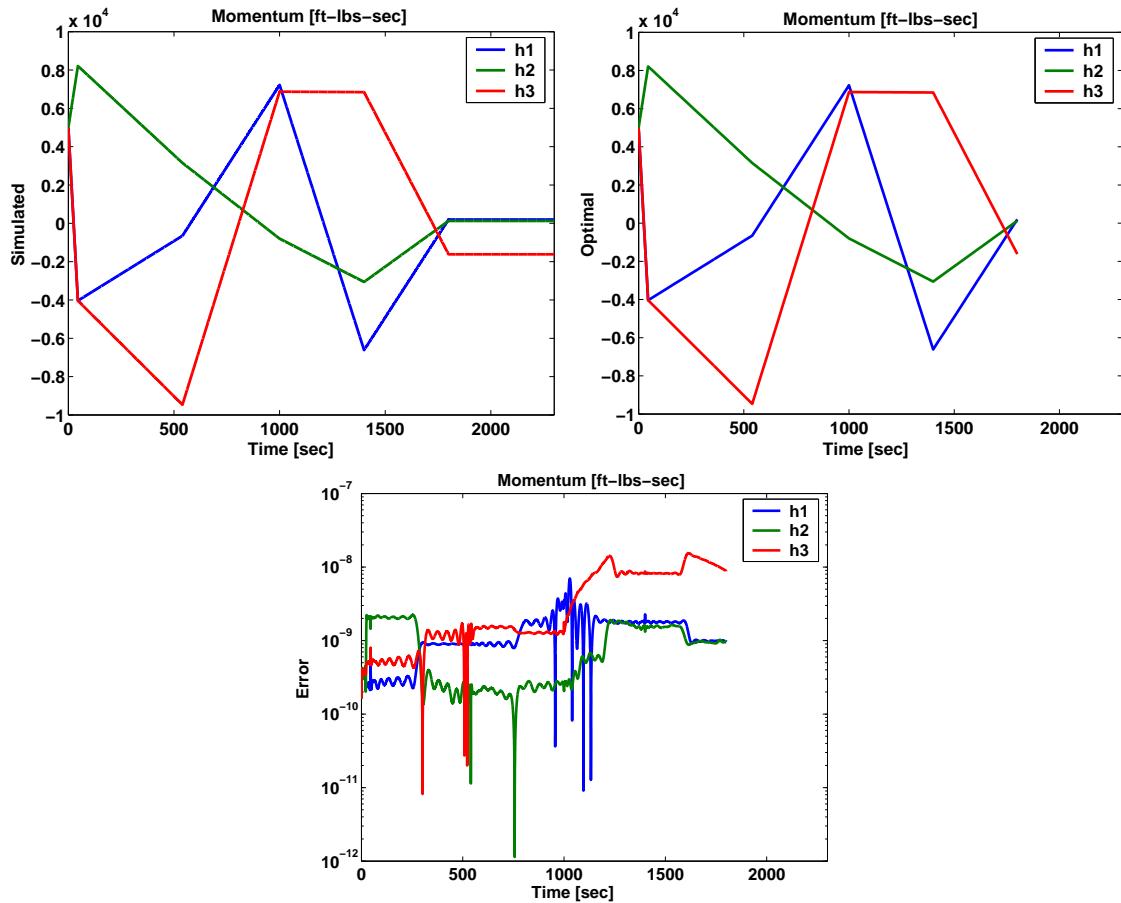


Figure 4.16: Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals

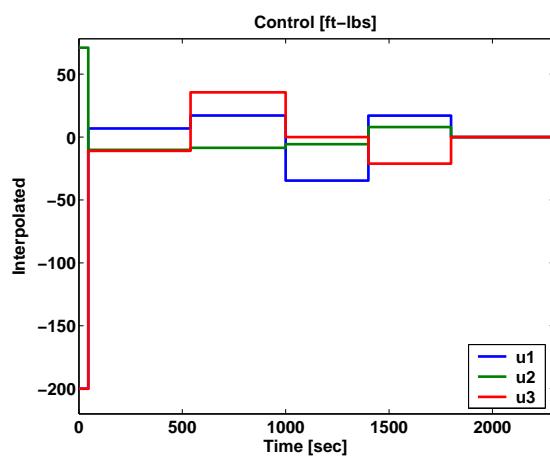


Figure 4.17: Computed Piecewise Constant Optimal Control, Extended by Zero for $t > 1800$, for Space Station Momentum Dumping Problem, Using $N=30$ on Five Subintervals

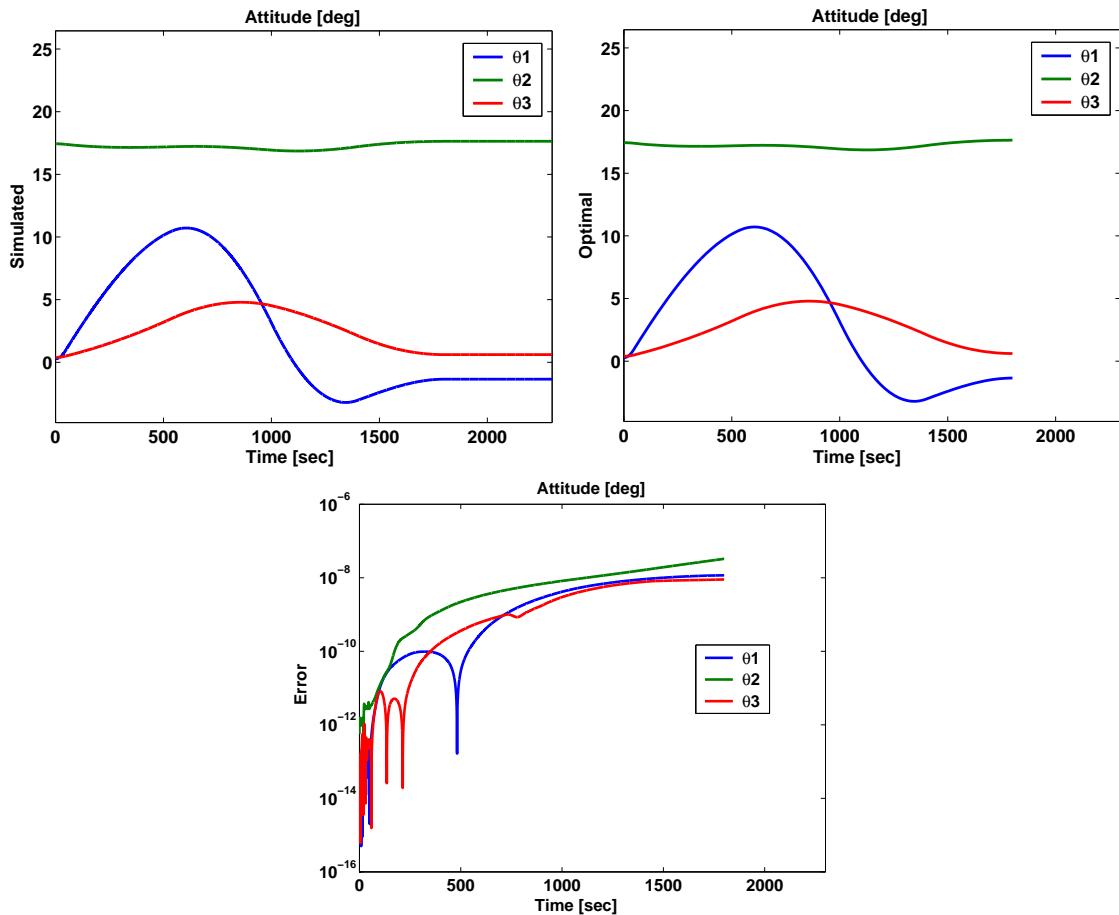


Figure 4.18: Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals

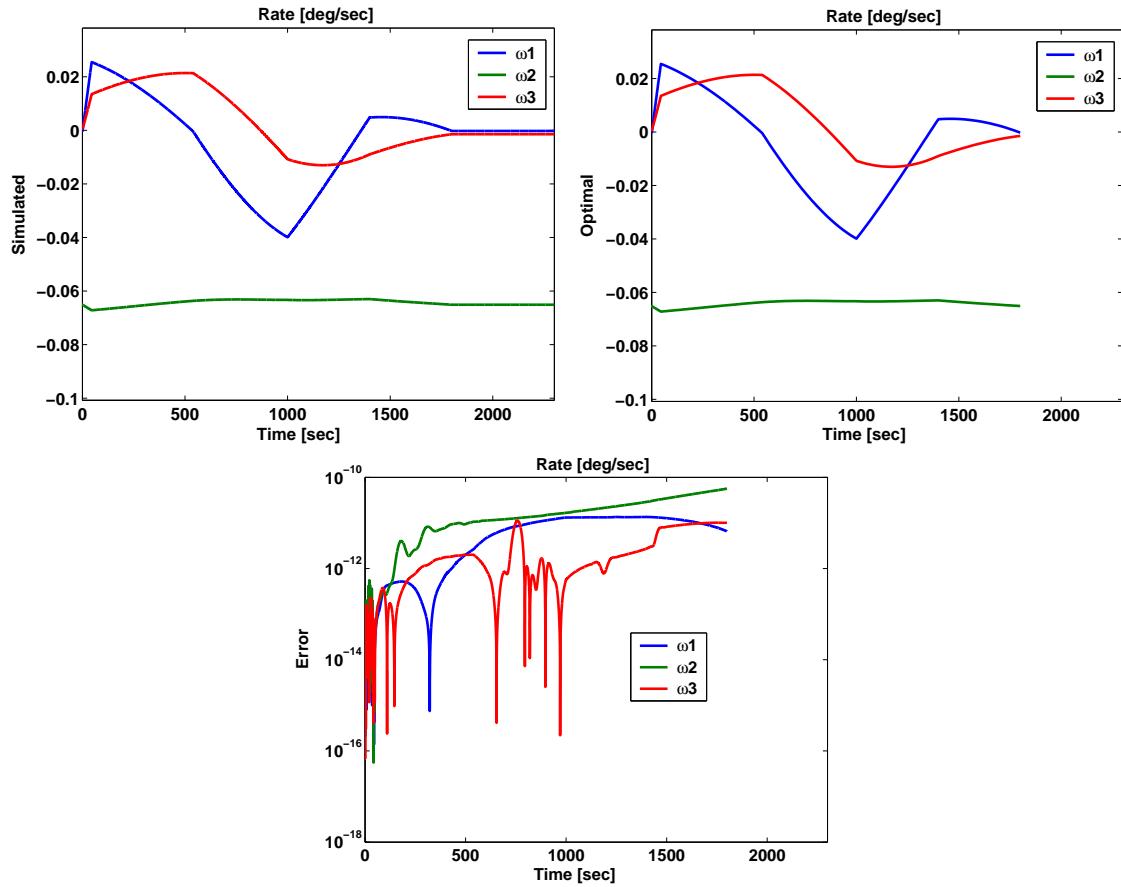


Figure 4.19: Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using $N=30$ on Five Subintervals

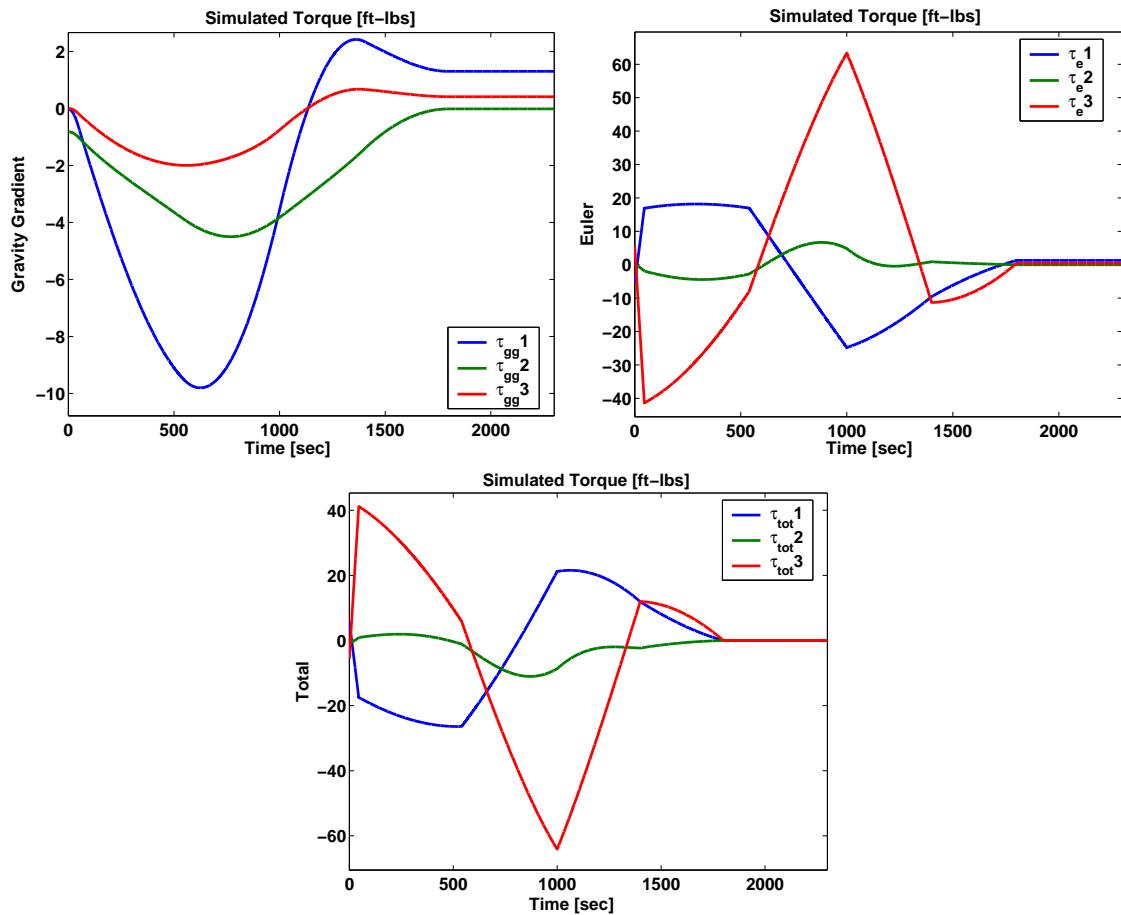


Figure 4.20: Simulated External Torques for Space Station Momentum Dumping Problem with Constrained Piecewise Constant Control, Using N=30 on Five Subintervals

4.5 ISS Momentum Dumping Problem with Control Law

This problem is a variation of the one described in Section 4.3 but with a control law introduced into the dynamics. Using a control law rather than a torque command, as done in Sections 4.3-4.4, allows the ISS to be controlled by an attitude command. Typically this control law is used to drive the spacecraft to some desired attitude by inputting an attitude command into the controller. This is an important variation to the momentum dumping problem because the ISS actually controls its attitude through this type of control law. Therefore, obtaining an optimal attitude command, as opposed to an optimal torque control, has more practical value. The control law used here is a proportional derivative control law which is described in [51]. This control law is formulated in terms of quaternions. The relationship between quaternions and Euler-Rodriguez parameters, which are used here, can be found in [32].

The state variables are the angular rate ω , the attitude r and the CMG angular momentum h . The control variable here is r_d instead of u due to the fact that the control for this problem is actually the desired attitude not the time derivative of momentum as in Section 4.3.

$$\min \quad ||h(t_f)||_2$$

s.t.

$$\begin{aligned}
J \frac{d}{dt} \omega(t) &= \tau_{gg}(r) - \omega(t)^\times (J\omega(t) + h(t)) - \frac{d}{dt} h(t), \quad t \in [t_0, t_f], \\
\frac{d}{dt} r(t) &= \frac{1}{2}(r(t)r(t)^T + I + r(t)^\times)(\omega(t) - \omega_o(r)), \quad t \in [t_0, t_f], \\
\frac{d}{dt} h(t) &= k_1 J \omega_{err}(\omega, r_d) + k_2 J r_{err}(r, r_d), \quad t \in [t_0, t_f], \\
||h(t)||_2 &\leq h_{max}, \quad t \in [t_0, t_f], \\
\omega(t_0) &= \bar{\omega}_0, \\
r(t_0) &= \bar{r}_0, \\
h(t_0) &= \bar{h}_0, \\
b(\omega(t_f), r(t_f), h(t_f)) &= 0,
\end{aligned} \tag{4.13}$$

where $r_d : \mathbb{R} \mapsto \mathbb{R}^3$ is the attitude command and

$$\begin{aligned}
C &= I + \frac{2}{1+r^T r} (r^\times r^\times - r^\times), \\
C_3 &= C \begin{pmatrix} 0, 0, 1 \end{pmatrix}^T, \\
C_2 &= C \begin{pmatrix} 0, 1, 0 \end{pmatrix}^T, \\
T &= C_3^\times (JC_3), \\
R &= I + \frac{2}{1+r_d^T r_d} (r_d^\times r_d^\times - r_d^\times), \\
R_2 &= R \begin{pmatrix} 0, 0, 1 \end{pmatrix}^T, \\
\omega_{err} &= \omega + \omega_{orb} R_2, \\
D &= CR^T, \\
r_{err} &= \frac{1}{2}(1 + D_{11} + D_{22} + D_{33})^{-1/2} \begin{pmatrix} D_{23} - D_{32} \\ D_{31} - D_{13} \\ D_{12} - D_{21} \end{pmatrix},
\end{aligned} \tag{4.14}$$

and

$$\omega_o = -\omega_{orb} C_2.$$

In (4.14), R_2 is the second column of the rotation matrix associated with the attitude control r_d . Again, b , $\bar{\omega}_0$, \bar{r}_0 and \bar{h}_0 are given by (4.7). The gains for the CMG controller k_1 and k_2 are given as

$$\begin{aligned}
k_1 &= 0.0632, \\
k_2 &= 0.002.
\end{aligned}$$

The control law described here forces the spacecraft to go to the desired attitude with a fixed angular rate. This is a restrictive model because real systems tend to

reach the required rate only at the end of the maneuver. However, this control law is consistent with the one used in [38] to control the same system.

MATLAB code for solving this problem can be found in Section B.3. Note that scaling factors were used to scale each variable in the optimization problem such that the constraint evaluations and variable magnitudes are of similar orders of magnitude. Values for the scaling factors used to solve this problem can be found in Section B.3. Results for this problem for $N = 150$ are shown in Figures 4.21 through 4.27.

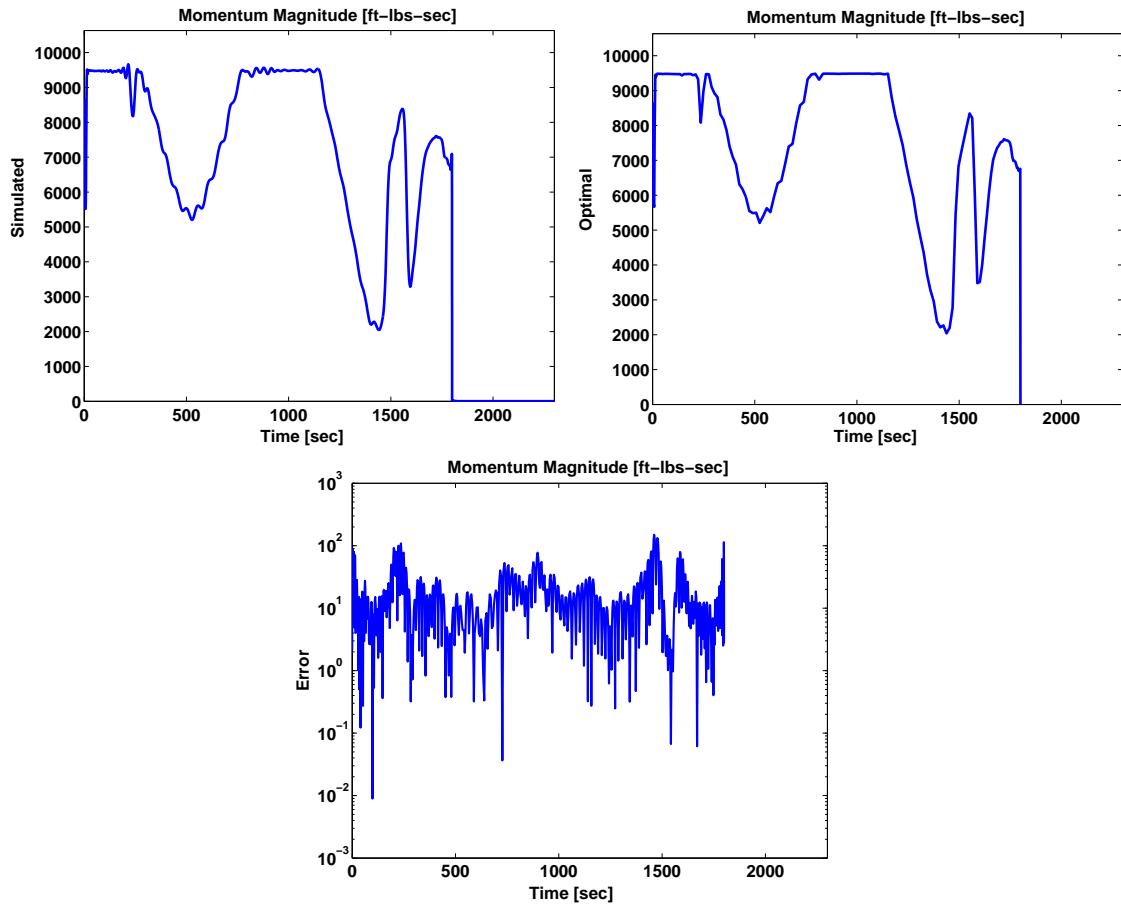


Figure 4.21: Simulated, Optimal and Error Angular Momentum Magnitude for Space Station Momentum Dumping Problem with Control Law, Using $N=150$

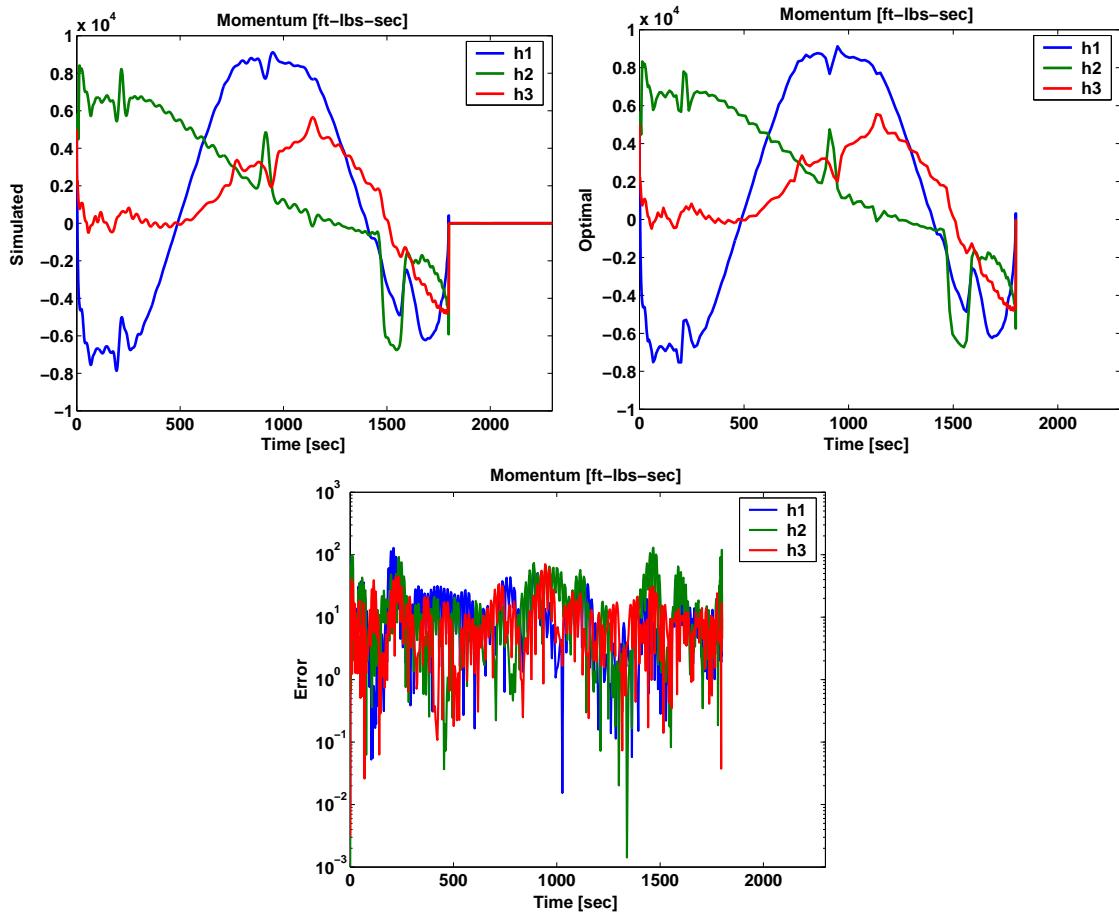


Figure 4.22: Simulated, Optimal and Error Angular Momentum for Space Station Momentum Dumping Problem with Control Law, Using N=150

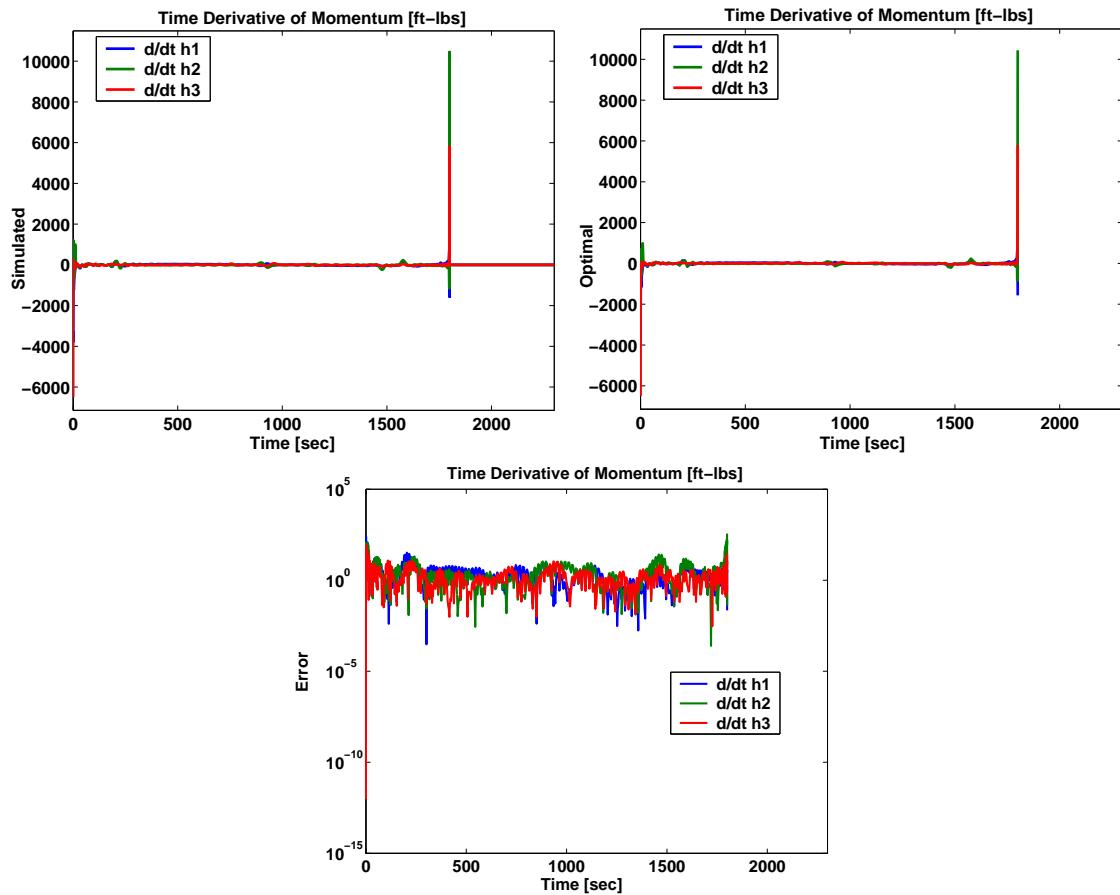


Figure 4.23: Simulated, Optimal and Error Angular Momentum Time Derivative for Space Station Momentum Dumping Problem with Control Law, Using N=150

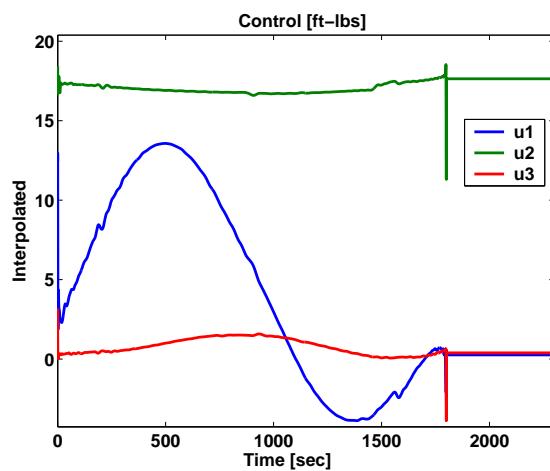


Figure 4.24: Computed Optimal Attitude Command Control, Extended by Zero for $t > 1800$, for the Space Station Momentum Dumping Problem, Using $N=150$

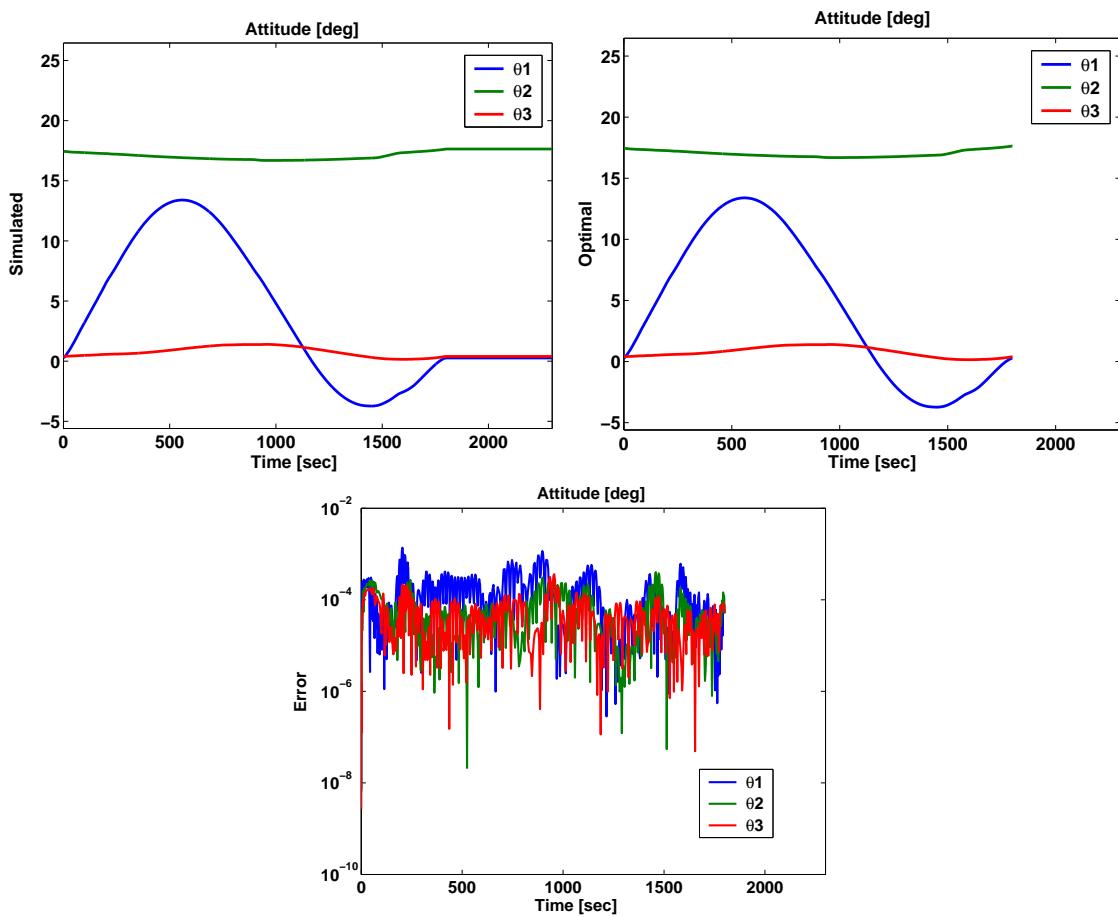


Figure 4.25: Simulated, Optimal and Error Attitude for Space Station Momentum Dumping Problem with Control Law, Using N=150

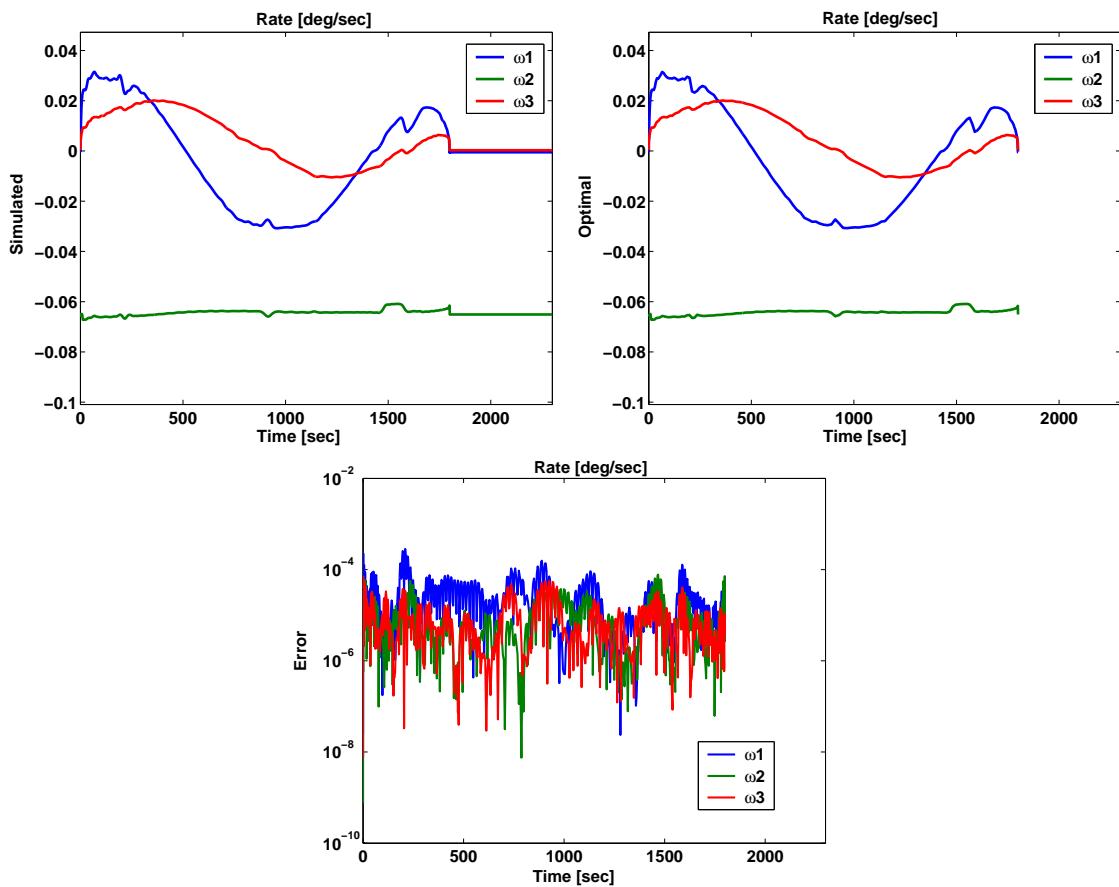


Figure 4.26: Simulated, Optimal and Error Angular Rate for Space Station Momentum Dumping Problem with Control Law, Using N=150

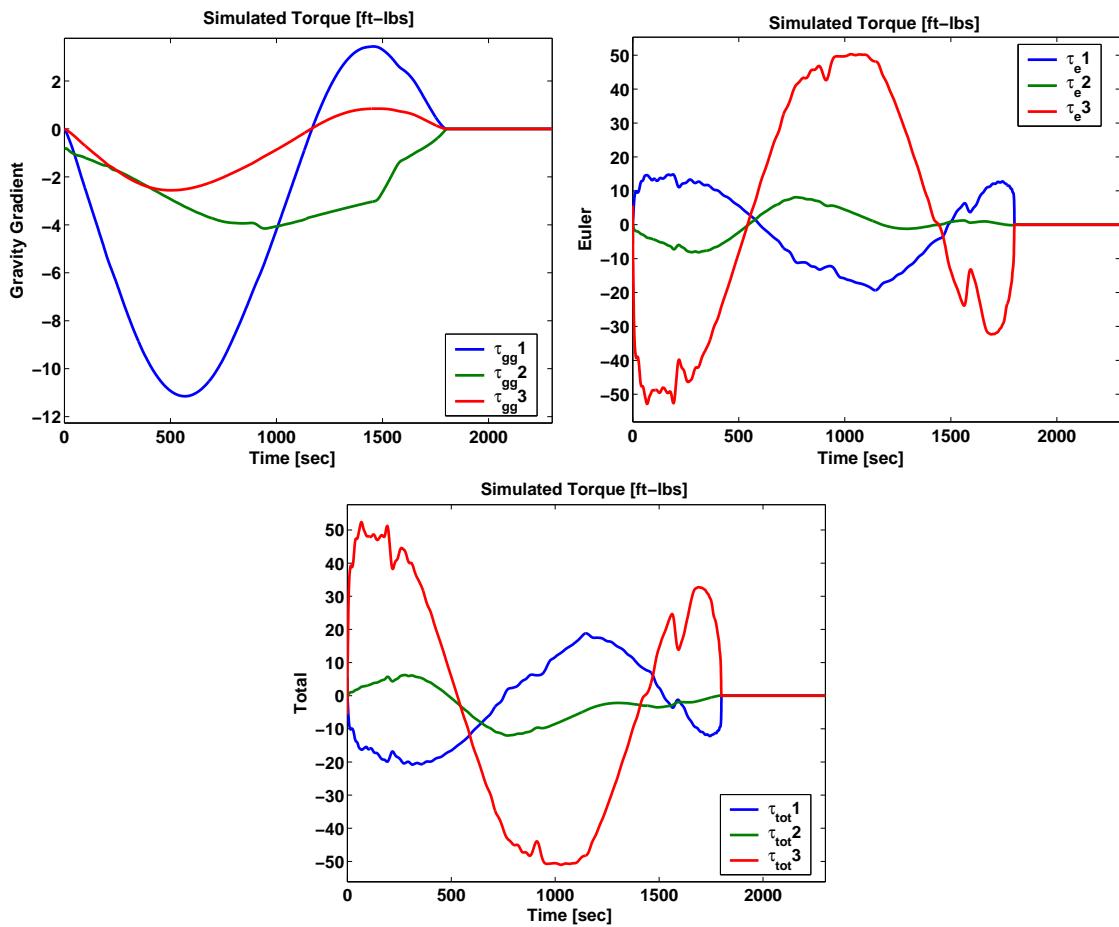


Figure 4.27: Simulated External Torques for Space Station Momentum Dumping Problem with Control Law, Using N=150

The results presented in this chapter represent a solution which is accurate in the sense that it has been verified through simulation, and practical in the sense that piecewise constant controls are implementable aboard the space station. This chapter demonstrated the utility of the Legendre Pseudospectral method for solving an optimal control problem. Due to the accuracy properties of this method, this problem was solved using few collocation points. The pseudospectral collocation method was also versatile enough to account for time dependent and piecewise constant control problems with little modification to the direct transcription of the optimal control problem.

Chapter 5

Conclusions and Future Work

In this thesis pseudospectral collocation methods for the direct transcription of optimal control problems were presented. It was shown that these methods exhibit properties which make it possible to relate the discretized NLP to the infinite dimensional OCP. This was done by constructing a new adjoint mapping that relates the Lagrange multipliers in the discretized optimal control problem to the adjoint variables corresponding to the infinite dimensional optimal control problem.

Through this adjoint estimation procedure, error estimates between the computed solution and the true solution to the optimal control problem were derived for linear-quadratic optimal control problems. It was shown that the conditioning of the optimality system matrix plays an important role in obtaining accurate solutions.

These methods were applied to the International Space Station momentum dumping problem to demonstrate their utility for difficult problems. The solutions obtained here are of great practical value as they may be directly implementable aboard operational spacecraft.

One suggestion for future work is to compute error estimates for the optimal control for nonlinear optimal control problems. This would be a significant result in the sense that many important applications are using pseudospectral collocation methods to solve optimal control problems. Such error estimate would give the scientists, engineers and mathematicians who use these methods more confidence in computed solutions.

Another suggestion for future work would be to develop a more robust optimal control solver that implements pseudospectral methods. Current packages either do

not give the user sufficient insight into the nonlinear programming solver, do not allow the user to supply analytical derivatives or do not allow the user to choose which pseudospectral collocation method is used.

A final suggestion would be to use pseudospectral methods, or another direct transcription method for that matter, to solve extensions of the ISS momentum dumping problem, as solutions to this problem can be quite useful in the aerospace industry. Extensions to the ISS momentum dumping problem include solving for a discrete (piece-wise constant) attitude command control that will remove built-up momentum, solving for an optimal control that is robust to unmodeled dynamics or changes in the boundary conditions, and attitude maneuvers during payload operations.

Bibliography

- [1] N. J. Adams and D. C. Redding. An Optimized Rotation Axis Model-Following Controller for STS Orbiter Vernier-Jet Attitude Maneuvers. Technical Report CSDL-R-1747, Draper Lab, 1984.
- [2] C. Ariotti. Space Online, http://www.ik1sld.org/iss_ft12a.htm, February 2003.
- [3] U. M. Ascher, R. M. M. Mattheij, and R. D. Russel. *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*. Classics in Applied Mathematics, Vol. 13. SIAM, Philadelphia, 1995.
- [4] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia, 1998.
- [5] N. Bedrossian, F. Ghorbel, and E. McCants. Space station momentum optimal cmg maneuver logic during payload operations. In *AIAA Paper 2000-4451*, 2000.
- [6] J. T. Betts. A survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 21:193–207, 1998.
- [7] J. T. Betts. *Practical Methods for Optimal Control using Nonlinear Programming*. Advances in Design and Control. SIAM, Philadelphia, 2001.
- [8] A.E. Bryson and Y.C. Ho. *Applied Optimal Control*. Hemisphere, New York, 1975.
- [9] R. Bulirsch, E. Nerz, H. J. Pesch, and O. von Stryk. Combining direct and indirect methods in optimal control: Range maximization of a hang glider. In R. Bulirsch, A. Miele, J. Stoer, and K. H. Well, editors, *Optimal Control –*

- Calculus of Variations, Optimal Control Theory and Numerical Methods*, pages 271–288. Birkhäuser, 1993.
- [10] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. Zhang. *Spectral Methods in Fluid Dynamics*. Springer Verlag, Berlin, Heidelberg, New York, 1988.
 - [11] C. Canuto and A. Quarteroni. Approximation results for orthogonal polynomials in Sobolev spaces. *Mathematics of Computation*, 38:67–86, 1982.
 - [12] R.K. Cavin, D. Colunga, and C.E. Hymas. Neighboring extremals for optimal control problems. *AIAA J.*, 11:1101–1109, 1973.
 - [13] S. S. Deshpande, M. L. Heck, R. R. Kumar, and H. Seywald. Minimum fuel spacecraft reorientation. *J. Guidance*, 17:21–29, 1994.
 - [14] G. Elnagar and M. Kazemi. Pseudospectral Chebyshev optimal control of constrained nonlinear dynamical systems. *Computational Optimization and Applications*, 11:195–217, 1998.
 - [15] G. Elnagar, M. A. Kazemi, and M. Razzaghi. The pseudospectral Legendre method for discretizing optimal control problems. *IEEE Transactions on Automatic Control*, 40:1793–1796, 1995.
 - [16] P.J. Enright and B.A. Conway. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. *J. Guidance, Control and Dynamics*, 15:994–1002, 1992.
 - [17] F. Fahroo and I. M. Ross. Costate estimation by a Legendre pseudospectral method. *Journal of Guidance, Control and Dynamics*, 24:270–271, 2001.
 - [18] F. Fahroo and M. Ross. A perspective on methods for trajectory optimization. AIAA/AAS Astrodynamics Conference, August 2002.

- [19] F. Fahroo and M. Ross. User's manual for DIDO 2002: A MATLAB application package for dynamic optimization. Technical Report NPS-AA-02-002, Naval Postgraduate School, Monterey, CA, June 2002.
- [20] F. Fahroo, M. Ross, and H. Yan. Real-time computation of neighboring optimal control laws. AIAA Guidance, Navigation and Control Conference, August 2002.
- [21] B. Fornberg. *A Practical Guide to Pseudospectral Methods*. Cambridge University Press, Cambridge, New York, 1996.
- [22] P. Fortescue and J. Stark. *Spacecraft Systems Engineering*. Wiley, New York, 1995.
- [23] G. Fraser-Andrews. A multiple-shooting technique for optimal control. *J. Optim. Theory Appl.*, 102:299–313, 1999.
- [24] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. Numerical Analysis Report 97–2, Department of Mathematics, University of California, San Diego, La Jolla, CA, 1997.
- [25] D. Gottlieb, M. Y. Hussaini, and S. A. Orszag. Theory and application of spectral methods. In R. G. Voigt, D. Gottlieb, and M. Y. Hussaini, editors, *Spectral Methods for Partial Differential Equations*, Philadelphia, 1984. SIAM.
- [26] W. Grimm and A. Markl. Adjoint estimation from a direct multiple shooting method. *J. Optim. Theory Appl.*, 92:263–283, 1997.
- [27] W. W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. *Numerische Mathematik*, 87:247–282, 2000.
- [28] M. Heinkenschloss. Optimization problems in computational engineering and science, CAAM 454 Course Notes. Rice University, 2002.

- [29] A.L. Herman and B.A. Conway. Direct optimization using collocation based on high-order Gauss-Lobatto quadrature rules. *J. Guidance, Control and Dynamics*, 19:592–599, 1996.
- [30] A.L. Herman and D.B. Spencer. Optimal, low-thrust earth-orbit transfers using higher-order collocation methods. *J. Guidance, Control, and Dynamics*, 25:40–47, 2002.
- [31] K. Holmstrom and A. Goran. *User's Guide for Tomlab v3.2.1*. Tomlab Optimization Inc., 2002.
- [32] P. C. Hughes. *Spacecraft Attitude Dynamics*. John Wiley and Sons, New York, 1986.
- [33] J. Jahn. *Introduction to the Theory of Nonlinear Optimization*. Springer Verlag, Berlin, Heidelberg, New York, second edition, 1996.
- [34] G. E. Karniadakis and S. J. Sherwin. *Spectral/hp Element Methods for CFD*. Oxford University Press, 1999.
- [35] A.Y. Lee. Neighboring extremals of dynamic optimization problems with path equality constraints. *J. Optim. Theory Appl.*, 57:519–536, 1988.
- [36] S. Liu and T. Singh. Fuel/time optimal control of spacecraft maneuvers. *J. Guidance*, 20:394–397, 1997.
- [37] The Mathworks Inc. *Using Matlab Version 6*, 5th edition, November 2000.
- [38] E. McCants. Optimal open-loop cmg maneuvers. Master's thesis, Department of Mechanical Engineering, Rice University, Houston, Texas, 2001.
- [39] MDA, S&DS, Space Station Division. Technical Description Document for the PG-1 Guidance, Navigation, and Control System. Technical Report MDC-95H0223D, McDonnell-Douglas, 1997.

- [40] Mission Operations Directorate Space Flight Training Division. International Space Station Familiarization. Technical Report TD-9702A, NASA, 1998.
- [41] H. J. Pesch. A practical guide to the solution of real-life optimal control problems. *Control and Cybernetics*, 23:7–60, 1994.
- [42] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, and E.F. Mishchenko. *The Mathematical Theory of Optimal Processes*. John Wiley & Sons, New York, 1962.
- [43] J.R. Rea. A Legendre pseudospectral method for rapid optimization of launch vehicle trajectories. Master’s thesis, Massachusetts Institute of Technology, 2001.
- [44] V. Schulz. Reduced SQP methods for large-scale optimal control problems in DAE with application to path planning problems for satellite mounted robots. Technical Report 96–12, Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Universität Heidelberg, 1996.
- [45] R. F. Stengel. *Optimal Control and Estimation*. Dover, New York, 1994.
- [46] O. Von Stryk and R. Bulirsch. Direct and indirect methods for trajectory optimization. *Annals of Operations Research*, 37:357–373, 1992.
- [47] R.A. Tapia. Optimization theory, CAAM 460 Course Notes. Rice University, 2002.
- [48] L. N. Trefethen. *Spectral Methods in Matlab*. SIAM, Philadelphia, 2000.
- [49] J. A. C. Weideman and S. C. Reddy. A MATLAB differentiation matrix suite. *ACM Transaction on Mathematical Software*, 46:465–519, 2000.
- [50] J. R. Wertz, editor. *Spacecraft attitude determination and control*. Kluwer Academic Publishers, Boston, 1995.
- [51] B. Wie. *Space Vehicle Dynamics and Control*. AIAA Education Series, Ohio, 1998.

- [52] L.J. Wood. Comment on neighboring extremals for optimal control problems. *AIAA J.*, 12:1452–1455, 1974.

Appendix A

DIDO: A Tool for Direct and InDirect Optimization

DIDO [19] is a MATLAB [37] based tool which implements the Legendre Pseudospectral method described in Section 3. The resulting nonlinear programming problem is solved using SNOPT [24] which is interfaced through MATLAB via an optimization tool called TOMLAB [31]. TOMLAB implements various optimization solvers, one of which is SNOPT, using MEX-files which call supporting Fortran routines. DIDO discretizes the infinite dimensional optimal control problem and sends the resulting NLP into TOMLAB to obtain solutions.

DIDO is set up to solve optimal control problems of the following form

$$\min E(y(t_0), y(t_f), t_0, t_f) + \int_{t_0}^{t_f} F(y(t), u(t), t) dt, \quad (\text{A.1a})$$

subject to the dynamic constraints

$$\frac{d}{dt}y(t) = f(y(t), u(t), t), \quad (\text{A.1b})$$

the event constraints

$$e_l \leq e(y(t_0), y(t_f), t_0, t_f) \leq e_u, \quad (\text{A.1c})$$

the path inequality constraints

$$h_l \leq h(y(t), u(t), t) \leq h_u, \quad (\text{A.1d})$$

and the state and control bounds

$$\begin{aligned} y_l &\leq y(t) \leq y_u, \\ u_l &\leq u(t) \leq u_u. \end{aligned} \quad (\text{A.1e})$$

In (A.1) we have used the notation applied in [19], which is slightly different from the one used in previous chapters. To use this tool, one must define at least two, up to four, auxiliary functions. An M-file which defines the objective function (A.1a). This file must, given state, control and time values, return the end-time cost function value E and the integral cost function value F at each collocation point. In addition, an M-file which defines the dynamics function must be defined. This file must, given state, control, time and state time derivative values, return the difference between the approximated state time derivative $\frac{d}{dt}y^N$ and the right hand side function f at each collocation point. The optional event function can be defined, which, given state and time initial and final values, return the value of the event function e . The optional path inequality function can be defined, which, given state, control and time values, return the value of the path function h . Bounds on each function, the states and the controls are passed directly to the DIDO function call. Examples of each function can be found in Section B.

DIDO is a relatively easy tool to use because the setup time involved in solving a given optimal control problem can be small as compared to most alternatives. The coding that is required of a user is minimal in the sense that only a few, possibly small, M-files need to be programmed. Additionally, the user interface, or DIDO function call, is very straight forward and resembles that of any standard MATLAB function call.

Despite these nice characteristics, there are some unattractive aspects to DIDO. The first is that, as the reader may have noticed, none of the user supplied functions output derivative information. Not only do they not require derivative information, the tool is not set up to allow the use of user supplied derivatives. Instead, finite difference derivatives are computed within SNOPT. This has several consequences. First and foremost, it is well known that the performance of an optimization algorithm can be severely slowed down when finite difference derivatives are used. As a consequence, solving relatively small problems can be very time consuming ordeal,

even when exact derivatives may be easy to compute. Secondly, the accuracy of solutions are impacted because the quality of derivative approximations are limited by the error in function values. In our context, error in function values are related to machine precision ϵ . For example, the error in derivative approximations, when forward finite differences are used, is on the order of $\sqrt{\epsilon}$ [28]. Therefore, solutions that are obtained by using DIDO may be less accurate than expected, especially if high order pseudospectral collocation discretizations are used.

Another disadvantage is that DIDO does not give the user the opportunity to control the NLP solver. Stopping tolerances, iteration limits and the like are all set within DIDO. As a consequence, accuracy expected from high order Legendre Pseudospectral collocation methods may be polluted by coarse optimization stopping tolerances. The user, while free to select the degree N of the Legendre Pseudospectral has no opportunity to adjust the accuracy of the NLP solve to match the expected discretization error.

Appendix B

MATLAB Code for the Solution of the ISS Momentum Dumping Problem Using DIDO

This Appendix contains the MATLAB code that may be used to produce the results for the ISS momentum dumping problem as solved by DIDO in Section 4.

B.1 ISS Momentum Dumping Problem with Continuous Control

The main file which calls DIDO is shown below.

```
%  
% issMain  
%  
% solve the ISS momentum dumping problem for  
% continuous control  
%  
% min ||h(tf)||  
%  
% s.t.  
% Jw' = Tgg(r) - w x (Jw+h) - u  
% h' = u  
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))  
% ||h|| <= 10000  
%  
  
%-----  
% Set global variables  
%-----  
global ws hs rs us w_orb J iJ  
  
% Scaling factors  
ws = 1e-3;  
hs = 1e3;
```

```

rs  = 1e0;
us  = 1e0;

% Orbital rate
w_orb = 0.06511*(pi/180);

% Inertia matrix
J = [2.80701911616000e+007 4.822509936000001e+005 -1.716750944480000e+007
      4.822509936000001e+005 9.514463934400001e+007 6.026044480000001e+004
      -1.716750944480000e+007 6.026044480000001e+004 7.659440133600001e+007];

% Inertia matrix inverse
iJ = [4.128859554604031e-008 -2.151370870617538e-010 9.254401181964391e-009
      -2.151370870617538e-010 1.051143985685158e-008 -5.648966425550106e-011
      9.254401181964391e-009 -5.648966425550106e-011 1.513006699674998e-008];

%-----
% Set input functions
%-----
iss.cost = 'issCost';
iss.dynamics = 'issDynamics';
iss.path = 'issPath';
iss.events = 'issEvents';

%-----
% Set variable parameters
%-----
t0  = 0;
tf  = 1800;
w0  = [-9.5380685844896e-006 -1.1363312657036e-003 5.3472801108427e-006]/ws;
wlb = inf*[-1 -1 -1]/ws;
wub = inf*[1 1 1]/ws;
wf  = w0;
h0  = [5e3 5e3 5e3]/hs;
hlb = [-1e4 -1e4 -1e4]/hs;
hub = [1e4 1e4 1e4]/hs;
r0  = [2.9963689649816e-003 1.5334477761054e-001 3.8359805613992e-003]/rs;
rlb = inf*[-1e10 -1e10 -1e10]/rs;
rub = inf*[1e10 1e10 1e10]/rs;
rf  = r0;

%-----
% Set time bounds
%-----
knots.locations = [t0 tf];

```

```

knots.definitions = {'hard', 'hard'};
knots.bounds.lower = [t0 tf];
knots.bounds.upper = [t0 tf];
knots.numNodes      = [150];

%-----
% Set variable bounds
%-----
bounds.lower.states(:,1) = [w0 h0 r0]';
bounds.upper.states(:,1) = [w0 h0 r0]';
bounds.lower.states(:,2) = [wlb hlb rlb]';
bounds.upper.states(:,2) = [wub hub rub]';
bounds.lower.states(:,3) = [wlb hlb rlb]';
bounds.upper.states(:,3) = [wub hub rub]';

bounds.lower.controls = -inf * [1 1 1]';
bounds.upper.controls = inf * [1 1 1]';

%-----
% Set constraint bounds
%-----
bounds.lower.path    = [0]';
bounds.upper.path    = [1e8 / hs^2]';
bounds.lower.events = 0 * [1 1 1 1 1 1]';
bounds.upper.events = 0 * [1 1 1 1 1 1]';

%-----
% Provide a guess
%-----
load iss_cont_guess;

%-----
% Call DIDO
%-----
[cost, primal,dual] = dido(iss, knots, bounds, guess);

```

The main file sets the dynamics function, the events function, and the path function for the DIDO tool. It also defines global variables, sets bounds on variables and function right-hand sides, and loads an initial guess from a data file. Note that weighting parameters were used to rescale each variable to be on the same order of magnitude. This procedure is discussed in [7, 19].

The function which implements the differential equation is shown below.

```

function [R] = issDynamics(primal)
%
% ISS momentum control dynamics
%
% Jw' = Tgg(r) - w x (Jw+h) - u
% h' = u
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))

% global variables
global ws hs rs us w_orb J iJ

% initialize variables
w = primal.states(1:3,:) * ws;
h = primal.states(4:6,:) * hs;
r = primal.states(7:9,:) * rs;
u = primal.controls * us;
N = length(w(1,:));
R = zeros(9,N);

for i = 1:N
    % compute auxiliary values
    C = eye(3) + 2/(1+r(:,i)'*r(:,i))*(skew(r(:,i))*...
        skew(r(:,i)) - skew(r(:,i)));
    C2 = C(:,2);
    C3 = C(:,3);
    wp = -w_orb * C2;
    Tgg = 3*w_orb^2*(cross(C3,(J*C3)));

    % compute differential constraint
    R(:,i) = primal.statedots(:,i) - [ iJ*(Tgg - ...
        cross(w(:,i),J*w(:,i)+h(:,i)) - u(:,i))/ws;
        (u(:,i))/hs;
        (0.5*(r(:,i)*r(:,i)' + eye(3) + skew(r(:,i)))*(w(:,i)-wp))/rs];
end

```

As discussed in [19] it is more computationally efficient to vectorize all functions. In order to do this, the above dynamics function must be expanded out component-wise. For the sake of brevity, this exercise is left out of this thesis.

The function which enforces the final time constraint is shown below.

```

function [E] = issEvents(primal)
%
% this function enforces that a TEA is reached
% at the final time by requiring that the ODE
% is zero at the final time when the control is zero
%

% global variables
global ws hs rs us w_orb J iJ

% initialize variables
w = primal.states(1:3,end) * ws;
h = primal.states(4:6,end) * hs;
r = primal.states(7:9,end) * rs;

% compute auxiliary values
C = eye(3) + 2/(1+r'*r)*(skew(r)*skew(r) - skew(r));
C2 = C(:,2);
C3 = C(:,3);
wp = -w_orb * C2;
Tgg = 3*w_orb^2*(cross(C3,(J*C3)));

% compute differential constraint
E = [ iJ*(Tgg - cross(w,J*w+h))/ws;
      (0.5*(r*r' + eye(3) + skew(r))*(w-wp))/rs];

```

The above function sets the constraint that a torque equilibrium attitude must be reached at the final time. As described in Section 4.2, this is equivalent to forcing the right-hand side of the differential equation to zero.

The function that implements the path inequality constraint is shown below.

```

function [P] = issPath(primal)
%
% this function enforces the path inequality constraint
% that ||h|| is less than or equal to 10000 at each time
%
```

```
% global variables
global ws hs rs us w_orb J iJ

% 2-norm-squared of h at each time
P = [(primal.states(4,:)*hs).^2 + (primal.states(5,:)*hs).^2 + ...
(primal.states(6,:)*hs).^2] / hs.^2;
```

The above function enforces the path inequality constraint on the norm of the angular momentum.

The function that implements the final time cost function on the norm of the angular momentum is shown below.

```
function [M, I] = issCost(primal)
%
% this function is the final time objective function
% on the 2-norm of h (squared)
%

% global variables
global ws hs rs us w_orb J iJ

% final time function
M = hs.^2 * primal.states(4:6,end)'*primal.states(4:6,end);

% no integral function
I = zeros(size(primal.states(1,:))) ;
```

B.2 ISS Momentum Dumping Problem with Piecewise Constant Control

The main file which calls DIDO is shown below.

```
%
% issMain
%
```

```

% solve the ISS momentum dumping problem for
% pw constant control
%
% min ||h(tf) ||
%
% s.t.
% Jw' = Tgg(r) - w x (Jw+h) - u
% h' = u
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))
% ||h|| <= 10000
%
%-----
% Set control bound
% 0 = no
% 1 = yes
%-----
bounded_control = 0;

%-----
% Set global variables
%-----
global ws hs rs us N1 N2 N3 N4 N5 J iJ w_orb

% N on each subinterval
N1 = 30;
N2 = 30;
N3 = 30;
N4 = 30;
N5 = 30;

% Scaling factors
ws = 1e-2;
hs = 1e3;
rs = 1e-2;
us = 1e0;

% Orbital rate
w_orb = 0.06511*(pi/180);

% Inertia matrix
J = [2.807019116160000e+007 4.822509936000001e+005 -1.716750944480000e+007
      4.822509936000001e+005 9.514463934400001e+007 6.026044480000001e+004
      -1.716750944480000e+007 6.026044480000001e+004 7.659440133600001e+007];

```

```

% Inertia matrix inverse
iJ = [4.128859554604031e-008 -2.151370870617538e-010 9.254401181964391e-009
      -2.151370870617538e-010 1.051143985685158e-008 -5.648966425550106e-011
      9.254401181964391e-009 -5.648966425550106e-011 1.513006699674998e-008];

%-----
% Set input functions
%-----
iss.cost = 'issCost';
iss.dynamics = 'issDynamics';
iss.path = 'issPath';
iss.events = 'issEvents';

%-----
% Set variable parameters
%-----
t0 = 0;
tf = 1800;
w0 = [-9.5380685844896e-006 -1.1363312657036e-003 5.3472801108427e-006]/ws;
wlb = [-1 -1 -1]/ws;
wub = [1 1 1]/ws;
wf = w0;
h0 = [5e3 5e3 5e3]/hs;
h1b = [-1e4 -1e4 -1e4]/hs;
hub = [1e4 1e4 1e4]/hs;
r0 = [2.9963689649816e-003 1.5334477761054e-001 3.8359805613992e-003]/rs;
rlb = [-1e10 -1e10 -1e10]/rs;
rub = [1e10 1e10 1e10]/rs;
rf = r0;

%-----
% Set time bounds
%-----
small = 100*eps;
knots.locations = [t0 360 720 1080 1440 tf];
knotsdefinitions = {'hard', 'soft', 'soft', 'soft', 'soft', 'hard'};
knots.bounds.lower = [t0 t0+small 540 1000+small 1400 tf];
knots.bounds.upper = [t0 540-small 1000 1400-small tf-small tf];
knots.numNodes = [N1 N2 N3 N4 N5];

%-----
% Set variable bounds
%-----
bounds.lower.states(:,1) = [w0 h0 r0]';

```

```

bounds.upper.states(:,1) = [w0 h0 r0]';
bounds.lower.states(:,2) = [wlb hlb rlb]';
bounds.upper.states(:,2) = [wub hub rub]';
bounds.lower.states(:,3) = [wlb hlb rlb]';
bounds.upper.states(:,3) = [wub hub rub]';

% note: parameters act as controls here,
% but the must be a control variable
% "placeholder" which is not iterated upon
if bounded_control == 0
    bounds.lower.parameters = -1e10 * ones(5*3,1);
    bounds.upper.parameters = 1e10 * ones(5*3,1);
elseif bounded_control == 1
    bounds.lower.parameters = -200 * ones(5*3,1);
    bounds.upper.parameters = 200 * ones(5*3,1);
end

bounds.lower.controls = 0;
bounds.upper.controls = 0;

%-----
% Set constraint bounds
%-----
bounds.lower.path = [0]';
bounds.upper.path = [1e2]';
bounds.lower.events = 0 * [1 1 1 1 1 1]';
bounds.upper.events = 0 * [1 1 1 1 1 1]';

%-----
% Provide a guess
%-----
load iss_dis_guess;

%-----
% Call DIDO
%-----
[cost, primal,dual] = dido(iss, knots, bounds, guess);

```

This main file does the same things as the one shown in Section B.1 except for two main things. The first is that the time interval is broken up into five subintervals so that a constant control can be used on each subinterval. This leads to the second

difference. In terms of the DIDO tool, the control in this setting is not used and the variable acting as the control is viewed as a parameter. DIDO must evaluate the control at every collocation point and it forces these variable to be variable in the optimization, even though the user may want the value to remain constant over each subinterval. The use of parameters alleviates this problem. Accomplishing the bound on the control can be accomplished by changing the parameter bound in the main driver file.

The function which implements the differential equation is shown below.

```

function [R] = issDynamics(primal)
%
% ISS momentum control dynamics
%
% Jw' = Tgg(r) - w x (Jw+h) - u
% h' = u
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))

% global variables
global ws hs rs us N1 N2 N3 N4 N5 J iJ w_orb

% initialize variables
w = primal.states(1:3,:) * ws;
h = primal.states(4:6,:) * hs;
r = primal.states(7:9,:) * rs;
u = reshape(primal.parameters,3,5) * us;
u = [kron(u(:,1),ones(1,N1)) kron(u(:,2),ones(1,N2)) ...
      kron(u(:,3),ones(1,N3)) kron(u(:,4),ones(1,N4)) ...
      kron(u(:,5),ones(1,N5))];
N = length(w(:,1));
R = zeros(9,N);

for i = 1:N
    % compute auxiliary values
    C = eye(3) + 2/(1+r(:,i)'*r(:,i))*...
        (skew(r(:,i))*skew(r(:,i)) - skew(r(:,i)));
    C2 = C(:,2);
    C3 = C(:,3);
    wp = -w_orb * C2;
    Tgg = 3*w_orb^2*(cross(C3,(J*C3)));
    R(1:3,i) = Tgg;
    R(4:6,i) = h;
    R(7:9,i) = wp;
end

```

```
% compute differential constraint
R(:,i) = primal.statedots(:,i) - [ iJ*(Tgg - ...
    cross(w(:,i),J*w(:,i)+h(:,i)) - u(:,i))/ws;
    (u(:,i))/hs;
    (0.5*(r(:,i)*r(:,i)') + eye(3) + skew(r(:,i)))*(w(:,i)-wp))/rs];
end
```

The events function, the path function and the cost function for this problem are the same as the ones stated in Section B.1.

B.3 ISS Momentum Dumping Problem with Control Law

The main file which calls DIDO is shown below.

```
%  
% issMain  
%  
% solve the ISS momentum dumping problem for  
% control law  
%  
% min ||h(tf)||  
%  
% s.t.  
% Jw' = Tgg(r) - w x (Jw+h) - u  
% h' = k1*J*w_e + k2*J*r_e  
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))  
% ||h|| <= 10000  
%  
%-----  
% Set global variables  
%-----  
global ws hs rs us k1 k2 J iJ w_orb  
  
% Scaling factors  
ws = 1e-2;  
hs = 1e3;  
rs = 1e-1;  
us = 1e-1;  
  
% Orbital rate
```

```
w_orb = 0.06511*(pi/180);

% Inertia matrix
J = [2.807019116160000e+007 4.822509936000001e+005 -1.716750944480000e+007
      4.822509936000001e+005 9.514463934400001e+007 6.026044480000001e+004
      -1.716750944480000e+007 6.026044480000001e+004 7.659440133600001e+007];

% Inertia matrix inverse
iJ = [4.128859554604031e-008 -2.151370870617538e-010 9.254401181964391e-009
      -2.151370870617538e-010 1.051143985685158e-008 -5.648966425550106e-011
      9.254401181964391e-009 -5.648966425550106e-011 1.513006699674998e-008];

% Controller gains
k1 = 0.0632;
k2 = 0.002;

%-----
% Set input functions
%-----
iss.cost = 'issCost';
iss.dynamics = 'issDynamics';
iss.path = 'issPath';
iss.events = 'issEvents';

%-----
% Set variable parameters
%-----
t0 = 0;
tf = 1800;
w0 = [-9.5380685844896e-006 -1.1363312657036e-003 5.3472801108427e-006]/ws;
wlb = [-1 -1 -1]/ws;
wub = [1 1 1]/ws;
wf = w0;
h0 = [5e3 5e3 5e3]/hs;
hlb = [-1e4 -1e4 -1e4]/hs;
hub = [1e4 1e4 1e4]/hs;
r0 = [2.9963689649816e-003 1.5334477761054e-001 3.8359805613992e-003]/rs;
rlb = [-1e10 -1e10 -1e10]/rs;
rub = [1e10 1e10 1e10]/rs;
rf = r0;

%-----
% Set time bounds
%-----
knots.locations = [t0 tf];
```

```

knots.definitions = {'hard', 'hard'};
knots.bounds.lower = [t0 tf];
knots.bounds.upper = [t0 tf];
knots.numNodes      = [150];

%-----
% Set variable bounds
%-----
bounds.lower.states(:,1) = [w0 h0 r0]';
bounds.upper.states(:,1) = [w0 h0 r0]';
bounds.lower.states(:,2) = [wlb hlb rlb]';
bounds.upper.states(:,2) = [wub hub rub]';
bounds.lower.states(:,3) = [wlb hlb rlb]';
bounds.upper.states(:,3) = [wub hub rub]';

bounds.lower.controls = -inf * [1 1 1]';
bounds.upper.controls = inf * [1 1 1]';

%-----
% Set constraint bounds
%-----
bounds.lower.path    = [0]';
bounds.upper.path    = [1e2]';
bounds.lower.events = 0 * [1 1 1 1 1 1]';
bounds.upper.events = 0 * [1 1 1 1 1 1]';

%-----
% Provide a guess
%-----
load iss_law_guess;

%-----
% Call DIDO
%-----
[cost, primal,dual] = dido(iss, knots, bounds, guess);

```

The above main file is similar to the one in Section B.1 because the control in this case is a continuous variable. The main distinction between this problem set-up and previous ones shows up in the differential equation. The function which implements the differential equation is shown below.

```

function [R] = issDynamics(primal)
%
% ISS momentum control dynamics
%
% Jw' = Tgg(r) - w x (Jw+h) - u
% h' = k1*J*w_e + k2*J*r_e
% r' = 0.5 * ( rr' + I + skew(r))(w - w(r))

% global variables
global ws hs rs us k1 k2 J iJ w_orb

% initialize variables
w = primal.states(1:3,:) * ws;
h = primal.states(4:6,:) * hs;
r = primal.states(7:9,:) * rs;
u = primal.controls * us;
N = length(w(1,:));
R = zeros(9,N);

for i = 1:N
    % compute auxiliary values
    C = eye(3) + 2/(1+r(:,i)'*r(:,i))*...
        (skew(r(:,i))*skew(r(:,i)) - skew(r(:,i)));
    C2 = C(:,2);
    C3 = C(:,3);
    wp = -w_orb * C2;
    Tgg = 3*w_orb^2*(cross(C3,(J*C3)));
    U = eye(3) + 2/(1+u(:,i)'*u(:,i))*...
        (skew(u(:,i))*skew(u(:,i)) - skew(u(:,i)));
    Ct = C * U';
    wt = w(:,i) + w_orb * U(:,2);
    pt = 0.5 * [Ct(2,3)-Ct(3,2);Ct(3,1)-Ct(1,3);Ct(1,2)-Ct(2,1)] /...
        sqrt(1+Ct(1,1)+Ct(2,2)+Ct(3,3));

    % compute differential constraint
    R(:,i) = primal.statedots(:,i) - [ iJ*(Tgg - ...
        cross(w(:,i),J*w(:,i)+h(:,i)) - u(:,i))/ws;
        (k1*J*wt + k2*J*pt)/hs;
        (0.5*(r(:,i)*r(:,i)' + eye(3) + skew(r(:,i)))*(w(:,i)-wp))/rs];
end

```